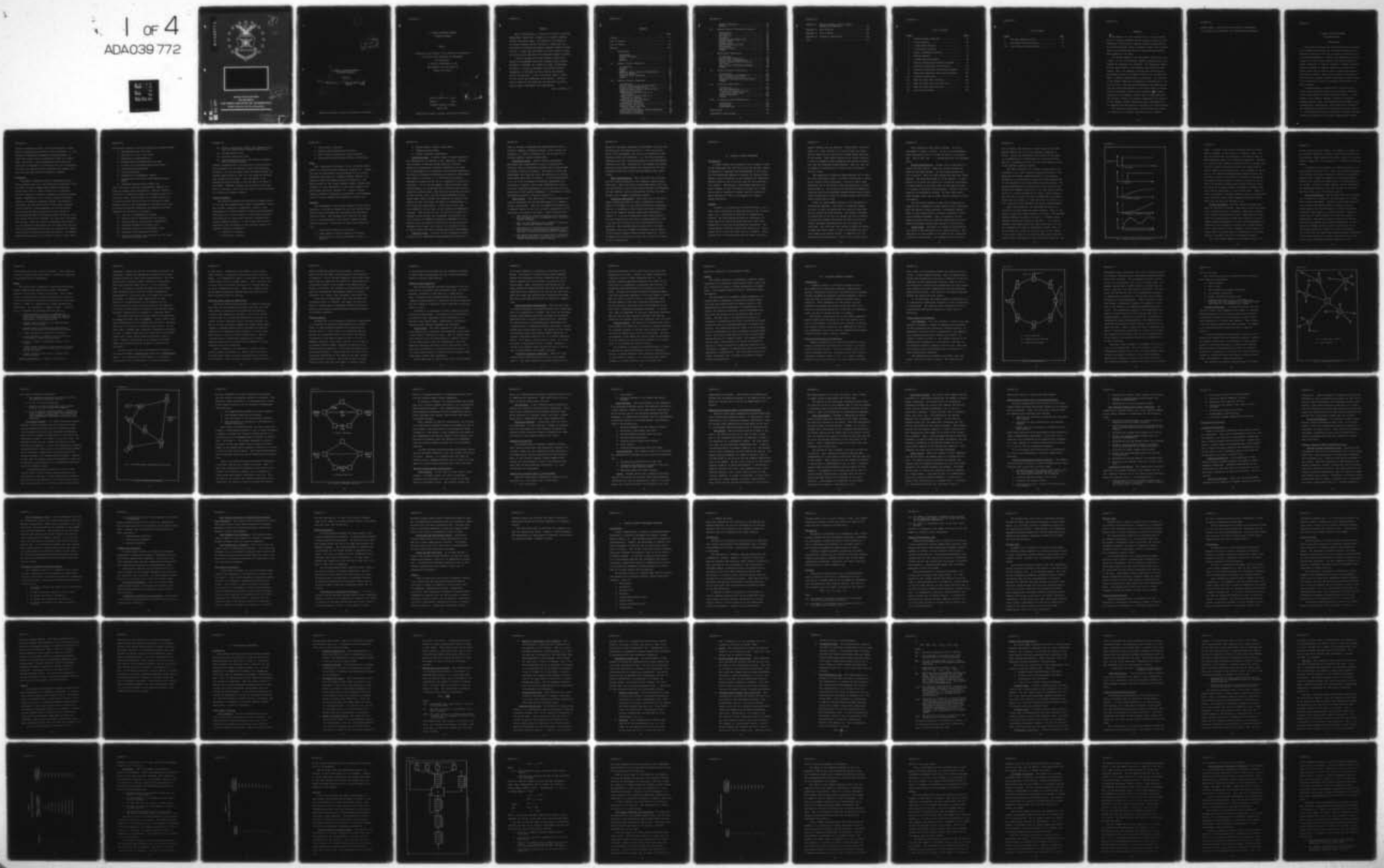
AD-A039 772   AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO   SCH--ETC   F/G 9/2
A GENERAL COMPUTER NETWORK SIMULATION MODEL.(U)
MAR 77   H M WARREN

UNCLASSIFIED   AFIT/GCS/EE/77-3   NL

1 OF 4
ADA039772



AF INSTITUTE OF TECHNOLOGY

UNITED STATES AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

1

6

A GENERAL COMPUTER NETWORK SIMULATION MODEL.

9 Masters THESIS,

14 AFIT/GCS/EE/77-3

10 Hoyt M. Warren, Jr.
Captain USAF

11 Mar 77

12 325p.

012 225

Approved for public release; distribution unlimited

GCS/EE/77-3

A GENERAL COMPUTER NETWORK SIMULATION MODEL

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by
Hoyt M. Warren, Jr., B.S.
Captain USAF
Graduate Computer Science

March 1977

Approved for public release; distribution unlimited

PREFACE

This investigation is a result of an effort to provide LGLA/AFDSDC, Gunter AFS, Alabama with a general computer network simulation model. The purpose of this model is to provide computer network analysts an additional tool with which to evaluate alternative computer network configurations. I hope that this model will prove useful to LGLA and to any others who want to investigate computer networks.

I wish to express my sincere thanks to Major Neal Morgan of LGLA/AFDSDC Gunter AFS, Alabama, Major Kenneth Melendez of AFIT/ENC, Mr. Richard M. Sowar of AFIT/SLSA, Mr. Wayne J. Jansen of ASD/ADOS, and Captain Richard C. Heidenreich of AFIT/ENE for their advice and guidance during this project. I wish to thank Dr. Gary B. Lamont for his advise and leadership as my advisor. Finally, I wish to thank my wife Wendy for her emotional and secretarial support throughout this undertaking.

Hoyt M. Warren, Jr.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The emphasis of this investigation is on the development of a general computer network simulation model for the evaluation of alternative computer network configurations. The simulation model allows a computer network and its workload to be characterized and then allows selected performance measures to be made.

The various phases of the development process are discussed. In the initial phase, symbolic system modeling and simulation techniques are surveyed. This discussion dwells heavily upon discrete-event modeling and simulation techniques. Next, the component architectures and essential performance measures of computer networks are investigated. Then, a computer network model is presented. The features of the network model are discussed and a description of job flow is given. This discussion describes the characterization and interrelationships of the various model variables and the algorithms governing their behavior. The implementation of the computer network model is then described and through a series of simulation experiments the behavior of the computer network simulation model (implementation) is compared with the behavior predicted by the network model.

The simulation experiments indicated that the simulation model was an accurate reflection of the computer network model. Several areas of potential improvement to the model and simulation are indicated and discussed.

A GENERAL COMPUTER NETWORK SIMULATION MODEL

I. INTRODUCTION

The growing complexity of United States Air Force base-level logistics functions, the increasing reliance of all logistics functions on the use of computer facilities, and the rapid advancements in computer and communications technology have led Air Force logistics systems developers to become interested in computer networks. Computer networks allow the sharing of computer resources on a much more global scale than is possible with a single maxi-computer time-sharing system. They are particularly interested in those areas of computer network investigation which promise to yield effective tools for analyzing and designing computer networks.

Although computer networks differ significantly in their construction and operation, the analysis and design of all types consist of three principle phases: 1. the design of the computer systems, interactive terminals, terminal control units, front-end processors, modems, data concentrators, multiplexers, and various other processing and communication devices (all collectively called nodes); 2. the topological analysis and design to specify both the capacity and location of each data communication channel (called transmission links), and the distribution of data bases and applications software within the network; 3. the design of transmission management software/hardware to ensure the orderly flow of information within the network (Ref 1:1385-1397). This investigation covers work done toward the development of an effective tool for the analysis and design of computer network topologies and computer network data base and software distribution schemes.

Background

The STALOG - BASE-TOP logistics automated data processing systems were to be the base-level corollaries to the Air Force Logistics Command (AFLC) Advanced Logistics System (ALS) (Ref 2). Where ALS was to be a wholesale oriented (supplier) logistics information processing system, STALOG - BASE-TOP was to be a retail oriented (consumer) logistics information processing system. As originally envisioned, the ALS would consist of six large, geographically dispersed, inter-connected, and on-line systems. The six data processing centers were to be located at Wright-Patterson AFB, Ohio; McClellan AFB, California; Kelly AFB, Texas; Tinker AFB, Oklahoma; Hill AFB, Utah; and Robins AFB, Georgia. Each site would be responsible for maintaining the inventories appropriate for the associated logistics center and additionally would have the capability of communicating with the other data processing centers. The STALOG - BASE-TOP systems were to be located at most Air Force Bases and would be required to meet the following functional needs:

1. Some Multiple-user data files.
2. Mobility of a unit in the field.
3. Modularity of implementation.
4. Geographically dispersed users per base.
5. A degree of inter-communication among users.
6. Some classified processing.
7. On-line data entry.
8. Availability of management reports.
9. Flexibility of functional needs among different bases.
10. Increased workload during system life.

[Two of the above functional requirements, mobility and security, deserve some clarification. Mobility refers to the ability to physically deploy the processing power of a given user from a fixed location to a remote location in a relatively short period of time (on the order of days). Security refers to the requirement of some users to be able to process classified information concurrently with other computing for periods up to 24 hours.]

11. Low risk of implementation failure.
12. Relative ease of implementation and testing.
13. Relative ease of used and user acceptability.
14. Minimized hardware and software cost.
15. Minimized personnel and development costs.
16. Minimized line/communications costs.
17. Low risk of failure, with relative ease of system recovery at minimal cost.

18. Minimal environmental impact (new/remodeled buildings, air conditioning, humidity control etc.).

19. Low maintenance costs.

20. Minimized operations costs.

21. Minimal conversion costs from existing automatic data processing systems.

Complete implementation of the ALS has not been achieved. The requirement for the extremely large data-base being accessed in an on-line network type environment pushed the current state-of-the-art (Ref 2:4) and cannot be achieved with present day technology. With the demise of the Advanced Logistics Systems, a new look had to be taken at the STALOG - BASE-TOP concepts and the ways in which they might be realized. One of the ways which is being evaluated is a base-level computer network.

Problem Statement

The Logistics Applications and Special Studies Branch of the USAF Data Systems Design Center (LGLA/AFDSDC), Gunter AFS, Alabama, expressed a pedagogical interest in the development of a general computer network system simulation which would aid in the evaluation of proposed topological configurations for base-level logistics information processing computer networks. It was desired that a system simulation allow evaluation of the following aspects of a computer network's performance:

1. Reliability (percent).
2. Availability (percent).

3. Survivability (percent).
4. Network effectiveness and efficiency.
5. Basic network configuration costs ($/month).
6. Data-base and applications software distribution.

Scope

This investigation discusses, first, the general development of a symbolic system simulation (Chapter II). Resulting from this discussion, are discussions on; the various components which make up a computer network (Chapter III), computer network performance measures (Chapter IV), the specifications of the general computer network system model (Chapter V), and a description of the actual computer network system simulation (Chapter VI). Finally, several example simulation experiments, conclusions, and recommendations for future research are discussed in Chapter VII.

Approach

The development of the general computer network system simulation, like the development of any other product, went through clearly definable but overlapping and frequently iterative steps. The following list indicates the steps taken in the development of the general computer network system simulation. A discussion of each step follows the list.

1. Study system simulation in general.
2. Study computer network components in general.
3. Study computer network performance measures in general.

4. Design computer network system model.

5. Implement system model.

6. Conduct simulation experiments.

Simulation Study. A general study of system simulation was conducted for the purpose of one, obtaining a rudimentary knowledge of general system simulation, and two, selecting an appropriate simulation technique to be used in the development of the computer network system simulation. The study of simulation techniques was limited to those which were symbolic in nature, i.e., those which are based on mathematical models and the use of a computer to perform the simulation. Of the various symbolic system simulation techniques studied, it was decided that a discrete event system simulation would be used. This decision was based upon three primary factors; first, the anticipated complexity of the system model itself; second, the necessity to represent the passage of time; and third, the degree of understanding of this technique versus others. It was also at this time that SIMSCRIPT I.5 was chosen as the simulation programming language. This was due to; one, its ability to handle the passage of time without real programmer intervention; two, its similarity to FORTRAN as opposed to SIMSCRIPT II.5 and SIMULA; three, the degree of familiarity of this simulation language versus other, and finally, its availability here at the Air Force Institute of Technology and at LGLA/AFDSDC.

Component Study. A general review of computer network literature (See supplemental bibliography) was conducted in order to develop a knowledge and understanding of what in actuality composes a computer network. Such a review was necessary as a partial foundation for the building of an accurate computer network system model.

Performance Study. Computer network performance measurement was investigated in order to determine the different types of performance measures which could be made on computer networks in general. The manner in which the measurements were made and their applicability to a general system model were also of interest. It was discovered that, as in the field of computer performance measurement and evaluation in general, there exist differences of opinion as to the essential measurements to make and even as to the definition of the measures themselves. A discussion of these performance measures is found in Chapter IV.

Model Design. This fourth step in the development process was concerned with the detailed design of a computer network system model. That is, the symbolic description of a computer network to the level of detail appropriate to the questions to be asked of the model. These questions were:

1. What degree of network reliability, availability, and survivability can be expected from a particular network topology?

2. What are the component costs ($/month) associated with a particular network topology?

3. What degree of effectiveness and efficiency can be expected from selected network components given a particular network workload and network topology?

4. How should data bases and applications software be distributed throughout a network for a particular network workload and network topology?

These four questions, supplied by LGLA/AFDSDC, were the sole motivation for the modeling process. It was during this phase of the development that the true architecture of the simulation model was described, i.e., the various network components and their interrelationahips and the performance measures to be used. Additionally, the program flow charts specifying each program module were prepared and the relationships between the various program modules were established.

Model Implementation. This fifth step was concerned with the actual production of the computer network system simulation. Major activities accomplished during this step were the coding of the various program modules, the debugging of these modules, and the intergration of these individual modules into a complete simulation program.

Simulation Experiments. This final step in the development process was taken in order to establish a certain degree of confidence in the system simulation's performance. That is, to establish to what degree the system simulation reflect the computer network system model described in Chapter V. A series of experiments were devised which it was felt exercised all the various program modules in such a way that any deviation from the system model would be observed. This experimentation process was by no means intended to serve as a formal verification and validation of the general computer network system simulation. Such a validation and verification effort was not within the scope of this investigation.

II. SYMBOLIC SYSTEM SIMULATION

Introduction

A system analyst has available to him a number of simulation techniques (Ref 3:425-447) which can provide a means of applying his knowledge and understanding of the system and its performance measures to satisfy his analysis needs. Symbolic system simulation is one technique for analyzing a system's behavior and is the one that will be discussed in this chapter. Specifically, this chapter describes the basic concepts of what constitutes a system, how a model can be used to represent a system, and how a system and model description effect the development of symbolic system simulation.

Systems

A system is a collection of related entities, (i.e., a job), each of which may be characterized by a set of attributes (i.e., job priority) which may in themselves be related. The relationships between the attributes of a given entity are called functions and quite often lend themselves to some form of mathematical expression. The relationships (functions) between the entities of a given system can be one of two types; static or dynamic. Static implies that the relationships does not change with time.

Dynamic implies just the opposite. Collectively, the attributes (i.e., their values) of an entity define its state, and the states of the entities of a system define the state of the system. These entity states can be viewed from both a static or dynamic system standpoint and provide a significant tool for examining system behavior. That is, they can be viewed at a single point in time or over a series of points in time.

The objectives in studying system behavior are to learn how state transitions occur, to predict transitions in state, and to control state transitions. One way in which these objectives can be satisfied is thru the evaluation of alternatives (Ref 4). An evaluation of alternatives approach is concerned with the relationship between system inputs, which induce transitions in state and system outputs which measure these transitions in state.

There are three common variants to the evaluation of alternatives approach. The first is a straight-forward analysis in which the system and its inputs are specified and the outputs are then measured. The second is broader in purpose and can be used to evaluate the relative merits of alternative system design when the input is given and certain desirable characteristics for the output are specified. The third and final variant is where the system is specified and an effort is made to determine the input that produces the desired output. A combination of the second and third approaches is used in this investigation.

Every system has three basic features. It has an environment in which it exists. It has a set of boundaries which distinguish the system from the rest of its environment. And it has a set of subsystems which are its component parts.

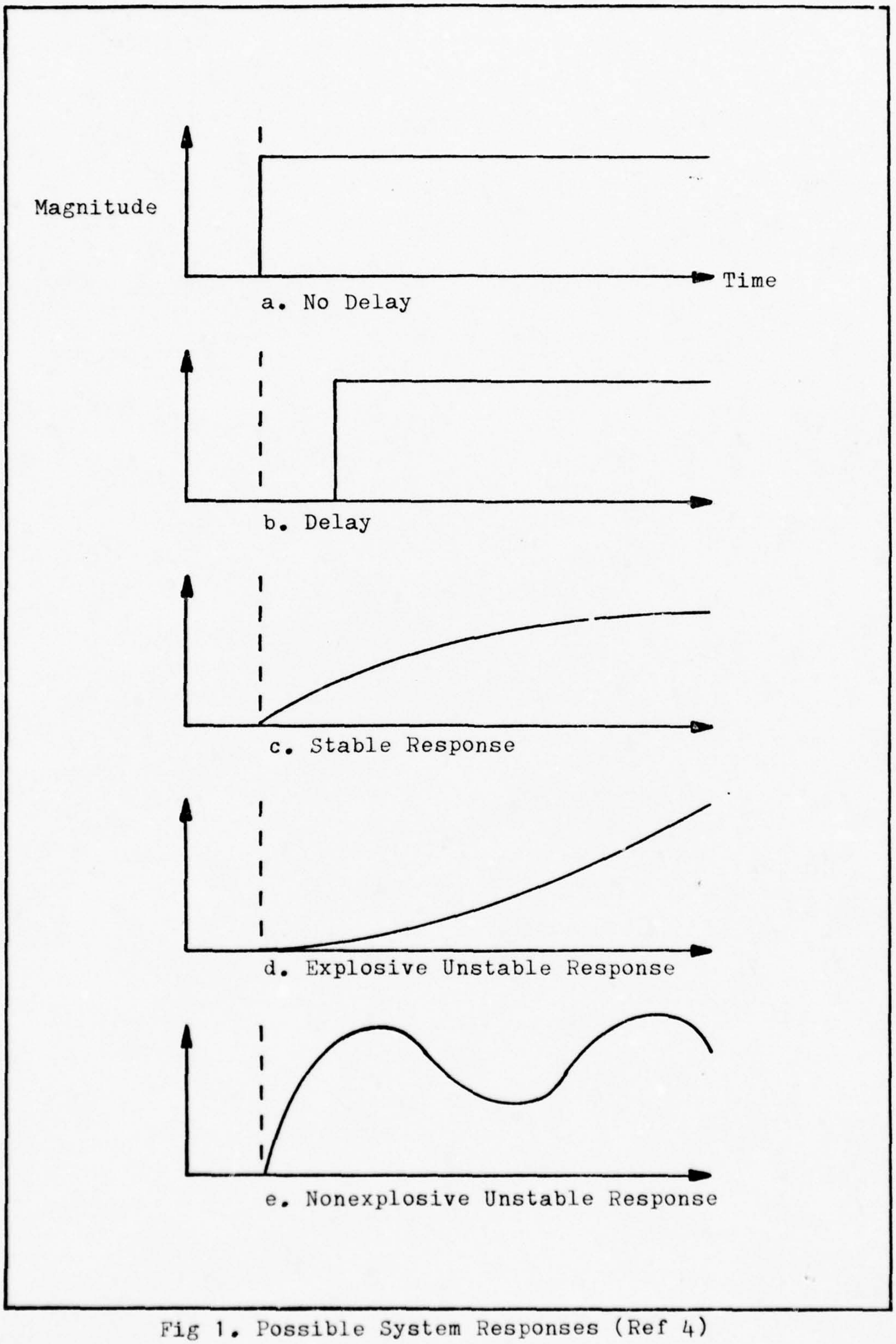
System Classification. Systems can be classified in a variety of ways (Ref 4). There are natural systems and there are man-made systems. The distinction between the two is obvious. There are open systems and there are closed systems. The distinction here is not so obvious. An open system is one which only exists in a particular environment. A closed system, on the other hand, is one which can exist in a number of alternative environments. Finally, there are adaptive and nonadaptive systems. As their names imply, these systems either can or cannot react to changes in their environment.

The distinction between an open and a closed and an adaptive and a nonadaptive system takes on real significance when the system is being analyzed. Conclusions reached about an open system must be carefully qualified in terms of the system environment. Also, the analysis of and adaptive system requires a complete description of how the system environment brings about changes in system state.

System State. The state of a system is determined by the values of the attributes of the system entities. The analysis of a given system is the study of its transitions in state as time elapses. If a system can be in a given set of states, then during any finite period of time the system exhibits one of a set of possible sequences of states. During any arbitrarily long period of time there is a probability of a system being in a particular state and also a probability of the current state changing to one of the possible remaining states.

The state transition of a system is described by two attributes (i.e., their values); magnitude and delay. The magnitude of a state transition refers to the absolute difference in an attribute's value over a specified period of time compared to its value before the state change. The delay associated with a transition of state refers to the elapsed time between the arrival of an input and the actual transition of state induced by the input. Delay reflects the fact that an input can cause a transition state immediately upon its arrival, at some time after its arrival, or over a period of time following its arrival (continuously).

When viewed together, the magnitude and delay in a transition of state describe system response. That is, the manner in which the system reacts to an input. Five possible system responses are shown in Figure 1. A stable system response is shown in Figures 1a, 1b, and 1c. In a stable response, the system moves over some finite period of time to a permanent new state of equilibrium (steady state) of finite value as a result of a single input to the system. If the initial value of the state variable under consideration is zero, the system is said to be in the empty or idle



a. No Delay

b. Delay

c. Stable Response

d. Explosive Unstable Response

e. Nonexplosive Unstable Response

Fig 1. Possible System Responses (Ref 4)

state. A system in the process of moving from one steady state to another is said to be in a transient state. The last two definitions are important in considering when to begin measuring output in a simulation experiment. Results, as a general rule, should be taken only after a steady state has been reached following the transition period from an initial empty or idle system state. This reduces the influence of the system's disproportionate behavior during the transition period on the overall evaluation of the system. An unstable system response is shown in Figures 1d and 1e. Figure 1d exhibits an explosively unstable response where the value of the state variable continues to increase with time without convergence to a new finite value. Figure 1e shows a nonexplosive unstable system in which an equilibrium level exists but where system response oscillates around this level but does not converge to it.

System Performance. Earlier, three objectives were given for analyzing a system. They were to learn how system state transitions occur, to predict transitions in state, and to control transitions in state. These specific objectives for system analysis result from a desire or need to improve system performance. In turn, system performance is reflected in the sequence of states that a system assumes over a given period of time. Normally, these sequences of system states are manipulated in some manner by the system analyst to provide a series of performance measures.

The idea of what composes a performance measure, of course, varies with the system. For example, in Chapter IV the essential measures of computer network performance will be discussed. These measures would have little or no significance if applied to the evaluation of a natural gas pipeline system.

In conducting the analysis of a complex system, it is often desired to obtain only a "feel" for system performance. It is exactly this type of analysis effort that can benefit greatly from a careful consideration of what performance measures will provide the needed "feel" or understanding of the system. Failure to devote adequate time to determining the performance measures to be used can lead to confusion and delay later on in the analysis process.

System Optimization. The ultimate objective in analyzing a system and in measuring its performance is to "optimize" system performance. Generally, this system optimization involves the identification of certain critical system aspects and the development of certain procedures to control these aspects. Usually, however, a number of these critical system aspects are beyond the analyst's observation. Such aspects impose constraints on system behavior and preclude total system optimization, in which case, the ultimate objective must be modified to one of optimizing performance subject to certain constraints. As is often the case with large complex systems, optimization of the entire system is virtually impossible, due to the sheer size of the system and the complexity of the relationships among the various subsystems. Thus, optimization can and must occur with regard to individual subsystems or selected groups of subsystems.

Models

The initial step in analyzing a system beyond defining system components and identifying possible performance measures, is to build a model of the system. Such a model can be a representation of theory concerning the system or a formal accounting of empirical observations of the system. Usually, a model is a combination of theory and empirical observation. The purposes of a model are many. In particular, it performs the following functions (Ref 4).

1. A model enables an analyst to organize his theoretical descriptions and empirical observations about a system and to deduce the logical implications of this organization.

2. A model leads an analyst to an improved understanding of the system.

3. A model brings into perspective the need for detail and relevance when describing the system.

4. A model expedites the analysis of a system.

5. A model provides a framework for the evaluation of alternative system configurations.

6. A model is easier to manipulate than an actual system.

7. A model permits control of more sources of system variation than the direct analysis of a physical system will allow.

8. A model generally costs less to analyze than a physical system.

Model Classification. A symbolic system model represents a system thru the use of mathematic equations and algorithms. Symbolic or mathematical models can be distinguished according to three major characteristics. The first characteristic separates symbolic models into analytical models and numerical models (Ref 4). In an analytical model it is possible to deduce the behavior of the system under analysis directly from the system's mathematical representation. Kirchoff's law is an example of an analytical model. A numerical model implies that an exact deduction of system behavior is not feasible but that numerical methods can provide descriptions of the behavior for certain system aspects as defined in the numerical model. Numerical integration is an example of a numerical model.

The second characteristic groups models into those that are static and those that are dynamic. A static model either omits consideration of the passage time entirely or describes the state of a system at a specified point in time. On the other hand, a dynamic model explicitly recognizes the passage of time. A dynamic model may also specify the relationships between the various system states at different points in time. A Markov process model is an example of a dynamic model that specifies both the passage of time and state dependence.

The third and final major characterization makes the distinction between a deterministic model and a probabilistic model. In a deterministic system model, all the system entities have fixed mathematical or logical relationships to each other. Consequently, the behavior of the system under analysis is completely determined by these relationships. In a probabilistic model, at least part of the entity relationships vary in a random fashion. Thus, an analyst can, at best, obtain an average description of a system behavior by using a probabilistic model. There are a number of different model descriptions possible when the three sets of model characteristics are combined.

Modeling Detail (Level of Observation)

During the model building process an analyst continually must deal with the problem of balancing the need for structural detail in describing the system with the modeling resources and capabilities available. By its very nature, a system model is a formalized abstraction of reality, of course, the more structural detail the model includes, the more it is felt the model resembles the actual system. Additionally, increased modeling detail provides an increased capability for observing system response to a given system modification or series of system modifications. That is, increased detail provides more combinations of system modifications which can be made and more aspects of system response which can be measured.

While it appears to be desirable to include as much modeling detail as possible in a model, there are several problems which result from this increased detail that must be considered. First, generally speaking, increased detail makes the modeling process more difficult. Second, it usually shifts the model characterization from analytical to numerical. Third, and most important, the analyst quite often does not understand the system to a degree that will allow specification of the system to the desired detail. Fourth, and finally, the inclusion of increased detail may place an increased and unacceptable demand on analysis resources, i.e., money, time, personnel, facilities, etc.

Every type of system model must limit the amount of structural detail it includes. The degree to which this detail is limited must be determined through a process of balancing the original system analysis objectives against the analysis resources.

Modeling Dangers

Although one of the purposes of building a system model is to improve the analyst's understanding of the system, there are three hazards associated with achieving this purpose (Ref 4). First, there is no guarantee that the results of the modeling effort will prove to be useful. Occasionally, this type of failure is due to a lack of adequate resources. More often, however, it is a case of an improper balance between available resources and the system analysis objectives. The second hazard deals with the analyst himself, in that he may come to think of his particular description of the system as the most accurate representation of reality. The third and most critical hazard involves the use of the model to analyze aspects of the system which the model was not intended to analyze. In other words to extrapolate the use of the system model without sufficient qualification.

Discrete Event Simulation

This section presents concepts applicable to the construction of a discrete event simulation model. An event denotes a transition in the state of a system entity. Thus, a discrete event simulation model can be defined as a model of system behavior in which entity state transitions are represented as a series of events occurring at discrete points in time.

Following is a discussion of event timing and entity/attribute relationships. The purpose of the discussion is to establish their importance in the realization of a discrete event system model. Additionally, alternative discrete event modeling techniques are discussed.

Event Timing.

The actual approach to the discrete event modeling of a particular system depends on the nature of the events' inter-arrival rates. That is, whether these inter-arrival rates are deterministic or random in nature. If the inter-arrival rates are deterministic, then the modeling techniques used must reflect inter-arrival rates which vary according to a fixed relation or are equal. If the inter-arrival rates are random, then the modeling techniques must reflect their randomness.

With either type of inter-arrival rate, the occurrence of an event signifies a transition in the state of the system. The states of system entities remain constant between the occurrence of events, therefore there is no need to account for this dead time in a discrete event system model. When a particular event occurs all the state transitions associated with the event are made, then simulated time is advanced to the time of the next event, where once again the required state transitions are made. This next event techniques allows the analyst to compress time.

Entity-Attribute Relationships. There are two types of primary structural relationships which play a significant role in the modeling of a system. The first are the mathematical relationships which exist between the attributes associated with the various system entities. Sometimes the specification of the mathematical relationships for a system serves to completely describe the manner in which system state transitions occur. The second are the logical relationships which exist between system components. A logical relationship checks to see if a certain condition holds. If it does, a given action is taken. If it does not hold, an alternative action is taken. Normally, a system is not modeled solely through the use of just one type of relationship, but through a mixture.

Alternative Modeling Techniques. There are three primary ways of building discrete event system models (Ref 5). The event scheduling technique emphasizes the detailed description of the steps that occur during the processing of an event. Usually, an event naturally has a distinct series of steps associated with it. The activity scanning technique emphasizes the review of all activities in a simulation to determine which can be initiated and terminated during the occurrence of a given event. Finally, the process interaction technique emphasizes the continuous progress of an entity through a system. That is, from its arrival event to its departure event. The development of the three techniques mentioned above has been directly associated with the development of discrete event simulation programming languages. In particular, GASP and SIMSCRIPT use the event scheduling technique; CSL uses the activity scanning technique; and GPSS and SIMULA use the process interaction technique (Ref 6:723-741).

Queueing Models. The majority of discrete event simulation models are reducible to a series of queueing problems. In a queueing problem, an arrival event occurs and causes an entity to demand that a service be performed. The system responds to the demand for service either by performing it or by keeping the entity waiting (places it in a queue) until it can perform the required service.

The objective in queueing oriented problems is usually to analyze how system performance varies in response to changes in system workload, system resource characteristics, or task selection schemes. In a given workload, system resources, and task selection must be resolved and explicitly specified in the simulation model.

Summary

The primary objective in developing a symbolic system simulation is to provide a system analyst with a tool which will allow him to analyze a system and measure its performance. The ultimate objective being to "optimize" system performance.

The development of a symbolic system simulation is a process which is composed of several well-defined steps. The first step requires that the analyst acquire a basic understanding of what components or subsystems make up a system, the analysis objectives, and the performance measures that are desired to be and should be made on the system. The next step is to describe the system as a collection of related entities each of which may be characterized by a set of attributes which may in themselves be related. This system description exists in an environment and is distinguished from this environment by system boundaries. The actual system description can be accomplished through the use of a symbolic model. This symbolic model can be either a representation of theory concerning the system or a formal accounting of empirical observation of the system. A symbolic or mathematical model can be either analytical or probabilistic. The final step involves the actual translation into a program for conducting system simulation experiments.

III. COMPUTER NETWORK COMPONENTS

Introduction

A computer network is any system composed of one or more computers and associated terminals, communication devices, transmission facilities, and specialized or general purpose transmission management software/hardware to facilitate the flow of information between terminals and/or computers. These components are the entities and subsystems which compose the network system and whose attributes and interrelationships must be defined and understood before any type of system analysis is possible.

The specific characteristics of the network components depend greatly upon the manner in which the computer network is implemented. This chapter will discuss, in general, the computer network topologies, the communication devices, the transmission facilities, and the transmission management software/hardware which compose a computer.

Network Architectures and Topologies

Computer networks can be classified according to their network architecture and topological structure. A network architecture has two basic levels; the global level of the overall computer networking approach and the local level of individual terminal network access. In a simple network, terminal access lines comprise the entire network structure.

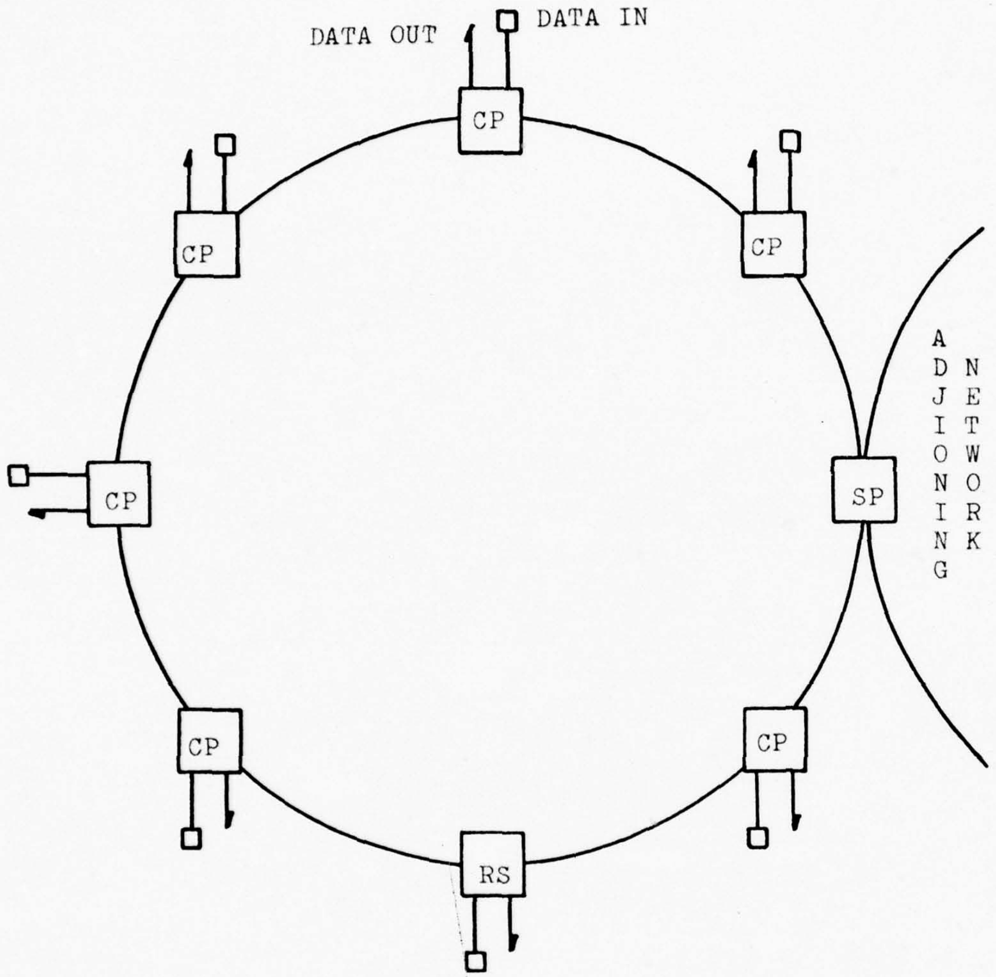
Thus, there is no difference between the global and local levels. In more complex networks, there are communication devices and transmission facilities in addition to the host computers and the interactive terminals. Some or all of the interactive terminals must communicate with the various host computers by first communicating with the associated communication devices. Thus there is a distinct difference between the global and local levels.

In the following, the global network architecture will be classified by topological structure as ring, centralized, or distributed. The terminal access architecture will be classified by topological structure as ring, star, or multipoint.

Global Network Architecture

Ring Topology. In a ring topology, a ring or loop type network is formed by a number of communication processors. The host processors and interactive terminals desiring to communicate with each other are connected to the communication processors and thus the global ring network. An example is shown in Figure 2. The primary function of the communication processors is to provide the interface for the terminals and host computers with the ring. More than one terminal and/or host computer may be connected to a given communication processor.

The communication processors cross their input and output lines with a shift register. The communication



RS - Ring Supervisor

CP - Communications Processor

SP - Switching Processor

Fig 2. A Ring Topology

processors switch information frcm input to output by shifting incoming information from transmission channels through their shift registers. The transmission channel capacity of the ring is multiplex into a series of time slots. For example, assume that a 10KBPS channel is divided into 20 slots. Each slot will then consist of 500 bits. The time slots flow through the ring from connumication processor to communication processor in one direction only (either clockwise or counter-clockwise). When a terminal or computer makes a transmission request, the request is first sent to and stored in its associated communication processor. The transmission request is then subdivided into blocks or packets which will fit into the time slots. A message header is attached to each block to indicate the transmission origin and destination. The communication processor then checks the shift register and waits for an empty time slot. When an empty time slot is detected, the block is shifted onto the channel to occupy the time slot. The communication processor also has the responsibility to detect which of the occupied time slots are addressed to it (Ref 7:527-530).

Sometimes a small computer is included in the ring to act as a ring supervisor. The ring supervisor acts as a transmission manager for the ring. A global network architecture can consist of several rings interconnected by another small computer acting as a switching processor. This switching processor controls the switching of messages from ring to ring.

The following are typical characteristics of networks with global ring topologies;

1. Easy to design.
2. Low in Cost.
3. High transmission channel throughput.
4. Low network reliability.
5. Higher transmission channel cost.
6. Possibly more suitable for inter-connecting terminals and host computers in a small geographic area than for a transcontinental network.

Centralized Topology. A centralized network can consist of a central host computer with a number of terminals connected directly to it to form a simple star structure. A centralized network can also consist of several simple stars connected to a central host computer. Thus, forming a far more complex centralized global network. An example of a centralized global network topology is shown in Figure 3.

A general transmission path between a terminal and a central host computer has the following sequence: terminal to terminal control unit, to multiplexer, to data concentrator, to front-end processor, to central host computer. While the above transmission path is what will generally be found in a structure of this type, not every centralized network or transmission path need contain terminal control units, multiplexers, or data concentrators.

The following are typical characteristics of networks

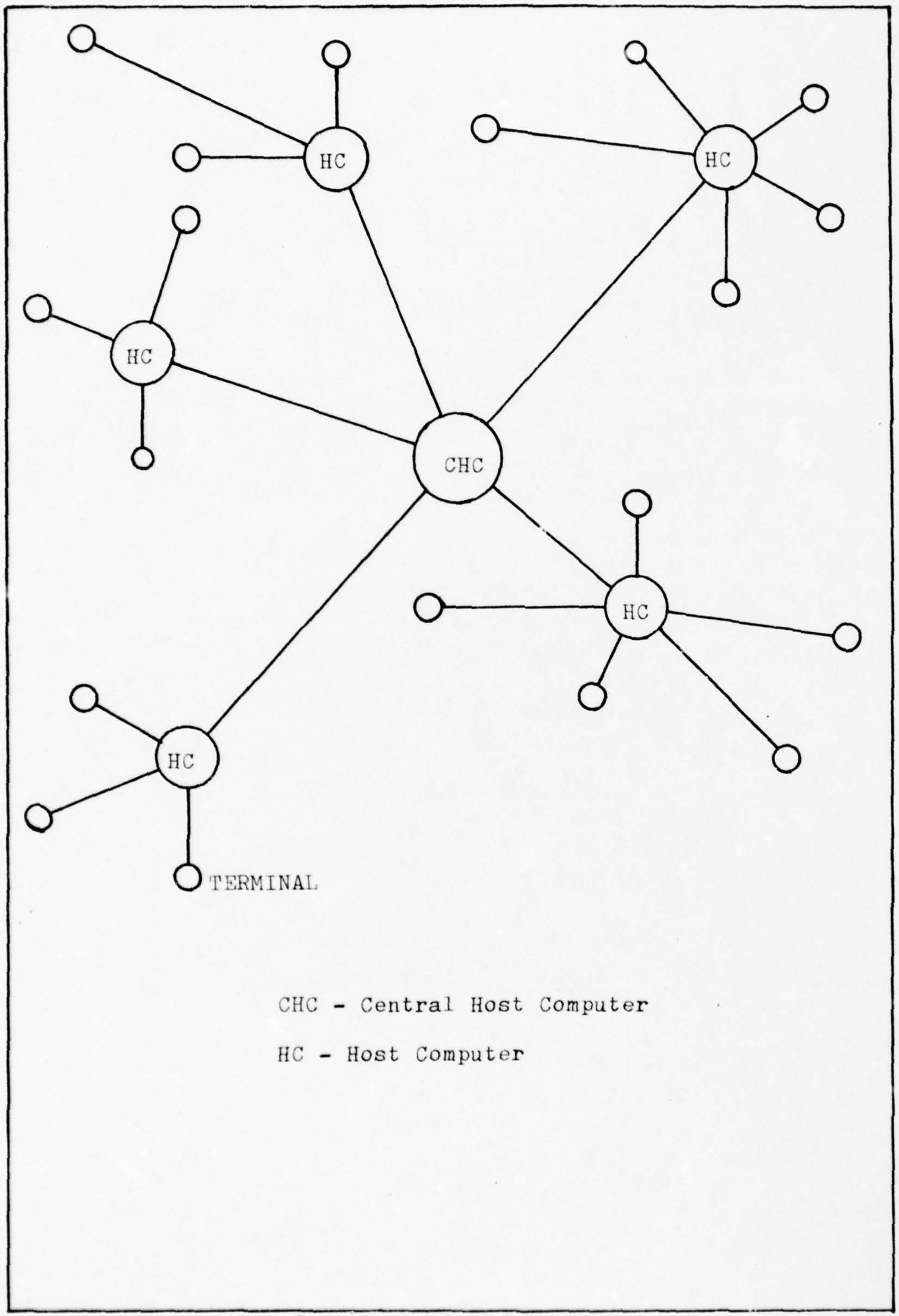


Fig 3. A Centralized Topology

with global centralized topologies:

1. The computing and message switching facilities are centrally located at one site.

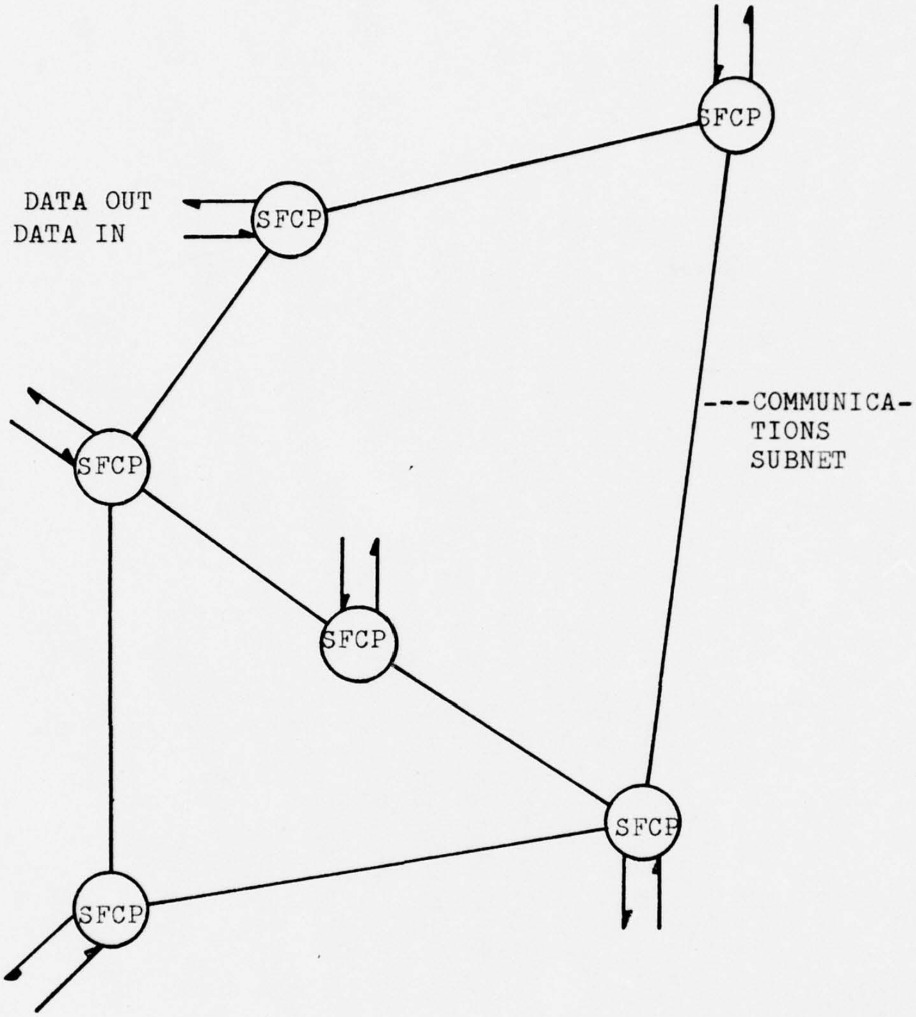
2. It has a tree-like appearance.

3. There is a unique transmission path between a terminal and its central host computer.

4. It is a terminal oriented network. Information flow is between a large number of terminals and their central host computers. There is little or no information flow between the central host computers.

Distributed Topology. In a global distributed network topology, several geographically distributed store-and-forward communication processors are connected together (either partially or fully) with dedicated transmission links to form a backbone network which is also called the communication subnet. See Figure 4 on the next page. The backbone network acts as a common user service to both terminals and computers. In order to communicate, terminals and computers must first access a store-and-forward communication processor. Messages are then sent through the backbone network by the communication processors. Networks of this type are normally classified according to their use of message or packet switching for transmission switching within the backbone network.

In message switching, messages are sent in their entirety along a predetermined fixed transmission path from source to terminal. At each communication processor along the path, the message is first stored on an on-line mass storage device or on an off-line mass storage device,



SFCP - Store-And-Forward Communications Processor

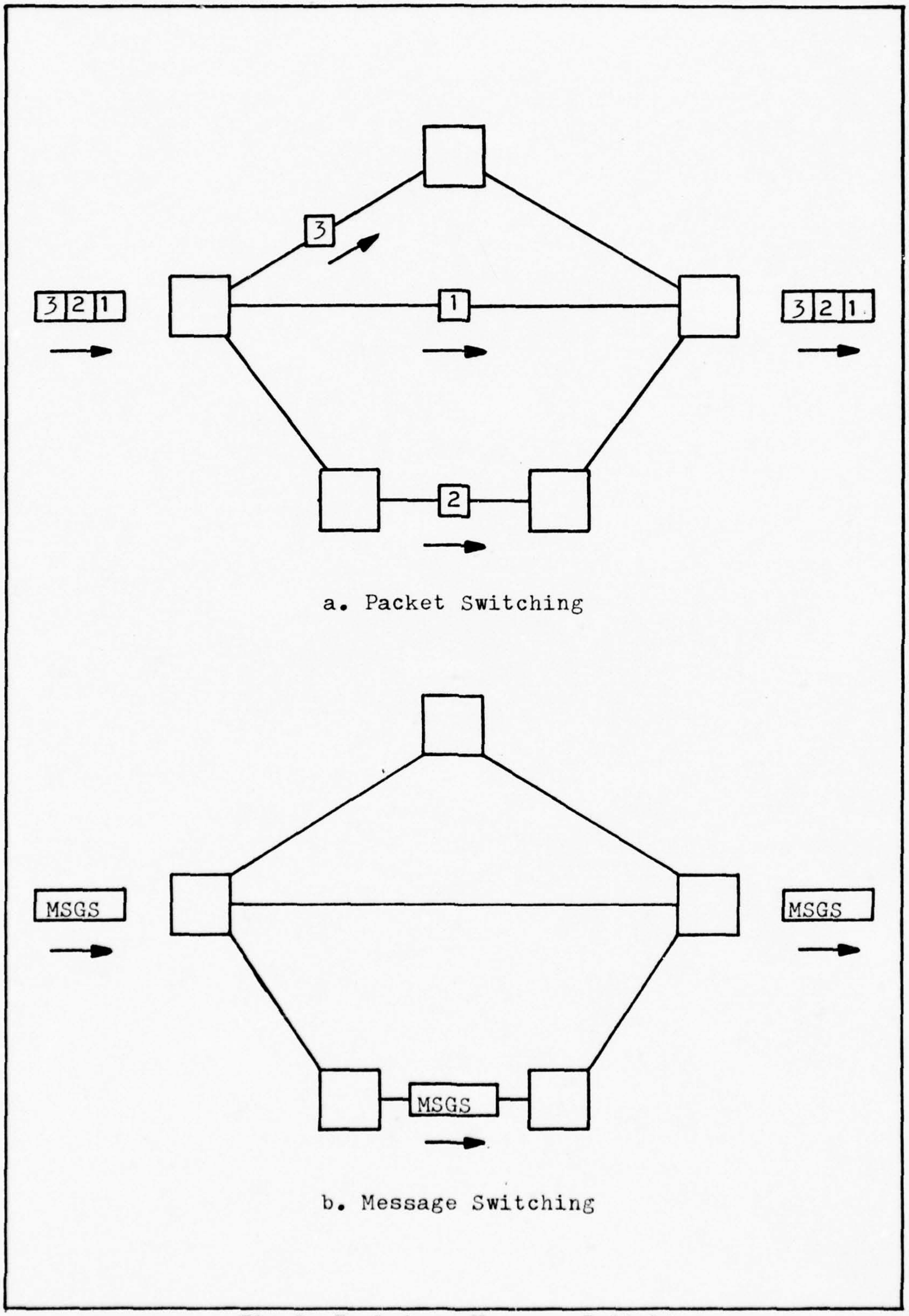
Fig 4. A Distributed Topology

and then forwarded to the next communication processor when the required transmission channel is available. Compared with packet switching to be discussed below, conventional message switching has the following disadvantages (Ref 8:271-309):

1. The communication processors are very expensive.
2. Long message delays can be incurred.
3. Less efficiency in utilizing network resources.
4. Less flexibility in adjusting to fluctuations in network conditions.

With regard to packet-switching, the basic difference between it and message-switching is that in packet-switching, a message is divided into frames or packets before it is transmitted and is reassembled when it reaches its destination. See Figure 5. The primary advantages of packet-switching is that the packets can be stored in the main memory of the communication processor, instead of an on-line or off-line mass storage device. This reduced substantially both the message delay time and the communication processor cost.

Many of the packet-switching's desirable characteristics result from the use of adaptive routing. This is where the transmission path between any two points in the network is not chosen in advance, but, is dynamically chosen based upon network conditions at the time. With this ability to allocate its resources based upon current conditions, the network is able to overcome the adverse



a. Packet Switching

b. Message Switching

Fig 5. Packet vs Message Switching

effects of temporary channel or communication device overload and failed backbone network components.

To transmit a message, the source precedes the message text with a terminal address and sends it to its associated communications processor. This communications processor then dynamically determines the best transmission path or route, provides error control, and notifies the source of the message receipt.

When a message is ready for transmission, the originating communication processor divides the message into one or more packets, each packet carries the appropriate header information. Each packet is then transmitted and makes its way independently through the network to the terminal communication processor, where the packets are reassembled into the original message and then sent to the intended terminal.

A packet-switched store-and-forward distributed network topology provides economical, fast, and reliable services to its users. However, it is advantageous over other approaches such as message-switching only if there is a large amount of traffic among widespread users (Ref 9:543-549).

Terminal Access Network Architectures

Ring Topology. In this type of local network architecture, a terminal control unit and its associated terminals are configured in a ring or loop in the same manner as described earlier. The terminal control unit acting as a communication processor interfaces with the ring of communication processors. IBM's 3600 system is an example of such an architecture (Ref 10).

Star Topology. A star terminal access architecture consists of a series of point-to-point connections. Every local access transmission line connects only one interactive terminal to a terminal control unit port, a multiplexer port, a concentrator port, and a computer port.

Multipoint Topology. A multipoint line has a tree-line appearance. In this topology, a number of terminals share one dedicated line; with terminals accessing the terminal control unit port either by contention or under the control of the terminal control unit itself.

Communication Devices

The classification of computer networks according their network architecture and topological structure also provides a basis for the classification and discussion of the communication devices associated with each topology. Thus, in this section, computer network communication devices will be discussed in terms of the topoloty with which they are most generally associated, i.e., ring, centralized, or distributed.

Communication Devices Used in a Ring Network

There are three major communication functions to be performed in a ring network (Ref 11:2947-2978):

1. Ring interfacing.

2. Ring control.
3. Switching between a ring network and another network.

Ring Interface. The ring interface is the communication processor mentioned earlier and basically consists of, a shift register, buffers and an associative store which can be written into by its connected terminals and host computers. A communication processor may be a minicomputer, a hard-wired device, or a microprocessor. The essential tasks it must perform are:

1. Breaking messages into time slot size blocks.
2. Detecting a usable empty time slot.
3. Shifting departing blocks onto the ring.
4. Detecting arriving blocks.
5. Shifting arrived blocks into buffers.
6. Error Control.
7. Erasing delivered blocks from ring slots.

Ring Controller. The essential tasks to be performed by a communication processor acting as the ring controller are:

1. Maintaining synchronization of the ring.
2. Preventing the build up of traffic in the ring because of undeliverable blocks.
3. Assigning empty slots when required.

Switch. Message blocks with a terminal destination outside a particular ring network have header addresses indicating this and are detected by a switch in the same manner that intra-ring blocks are detected by the ring communication processors. These blocks are buffered and shifted onto the adjoining network in the same way that ring blocks are shifted onto a ring by the communication processors.

Communication Devices Used in a Centralized Network

There are three major communication devices associated with a centralized computer network: a multiplexer, a data concentrator, and a front-end processor. Additionally, there are several other communication devices which can be used in a centralized network and which will be discussed here.

Multiplexer. Two definitions need to be made at this point. The term facility will be used to refer to that part of the telephone plant which is described in terms of its properties as a transmission medium. The term channel will refer to any functional transmission path. A channel is characterized by its channel capacity, i.e., the maximum rate at which information can flow through the channel. The channel capacity depends on a number of factors, such as the band width of the facility and the hardware characteristics of the modems. The use of one facility to form several separate channels is called multiplexing. A device which combines multiple facilities, each possessing one or more distinct channels, into a single facility with the same distinct channels, is called a multiplexer. A device which performs the reverse process is called a demultiplexer. Many current devices perform multiplexing in one direction, and demultiplexing in the other direction. Such a device is simply called a multiplexer (Ref 12:161-168).

The channel is the functional transmission path, where the facility is part of the hardware used to form the channel. A multiplexer does not change the channel structure of the network. Thus, the multiplexer is functionally transparent.

Data Concentrator. Consider a communication device having several facilities connected to its input, and only one facility connected to its output. From this description, the device might be a multiplexer. However, it is distinguished from a multiplexer by the following characteristics: the single facility on the output side of the device carries only one channel, the capacity of which is less than the sum of all of the channel capacities on the input side of the device. Such a communication device is called a data concentrator (Ref 13:1313-1321).

The percent of time a channel is in use is called its utilization. Most interactive terminals generate data transmissions at a rate which is much less than the capacity of the channel. Thus, resulting in low channel utilization. A concentrator is advantageous to use, in that it replaces several under utilized channels with one highly utilized channel. A prerequisite for concentrator use is that its output channel capacity must be greater than the sum of the average data rates of the terminals on its input channels. A multiplexer is transparent to the channel structure of a network; a concentrator is not.

Front-End Processor. The central host computer and the terminals of a centralized network use the network to interchange information. The general means of transferring information between the central host computer and the terminals and the computers connected to it, is thru its input/output channels. Certain devices, such as printers, disks, tape units, etc., are connected to the central computer with a hardwired interface. In the case of a communications link, the modem which terminates the link must be interfaced with the central computer. The overhead required for the central computer to itself interface with many communication link modems is too great to be economically feasible. Thus, a sophisticated interface (Front-End Processor) is used to handle the model interfacing, and only essential information is transferred through one of its input/output channels to the central host computers. Minicomputers are often used as Front-End Processors.

Other Devices. There are a number of other communication devices which are commonly used in a centralized network. The modem is used to interface the computer and terminal digital baseband requirement with the voice-grade telephone line analogue bandpass requirement. The modem sharing unit is a device for connecting several terminals to a single modem. A port sharing unit is a communication device for connecting several modems to a single multiplexer, data concentrator, or a computer port. A biplexer is a device which utilizes two voice-grade lines to effectively achieve a single high speed channel.

Communication Devices Used in Distributed Networks

The basic communication devices used in a store-and-forward packet-switched distributed network are the communication processors. There are three major functions which the communication processors must perform:

1. Interface the interactive terminals with the backbone network.
2. Interface the host computers with the backbone network.
3. Manage the flow of information through the backbone network.

These three functions can be performed by three distinctive types of communication processors. One or more of the functions can be performed by one communication processor. Or, a host computer or a remote concentrator may perform all or part of the two interfacing functions.

In the following, each function will be discussed as if it were being performed by a distinct communication processor.

Terminal-Communication Subnet Interface. The communication processor serving as a terminal-communication subnet interface must perform the following tasks:

1. Recognizing functional message applications, such as file transfers, remote-job-entry, etc., so that appropriate protocols can be applied.
2. Dividing outgoing messages into packets.
3. Formating the message packets.
4. Code-converting for a variety of different types of terminals.

5. Attaching the message packet headers and trailers.

6. Storing of unacknowledged message packets for possible retransmission.

7. Reassembling of received message packets.

Host Computer-Communication Subnet Interfaces. The communication processor serving as a host computer-communication subnet interface is responsible for the following tasks:

1. Dividing outgoing messages into blocks which are compatible with the network protocol.

2. Formatting and code-converting the message blocks into a standard format and code acceptable to the network.

3. Attaching the message block headers and trailers.

4. Storing the unacknowledged message blocks for possible retransmission.

5. Reassembling received message blocks into messages.

6. Dividing outgoing message blocks into packets which are acceptable to the network protocol.

7. Attaching the packet headers and trailers.

8. Storing the unacknowledged packets for possible retransmission.

9. Reassembling received packets into message blocks.

10. Controlling the message input rate to avoid overload.

Information Flow Manager. The communication processor serving as the network information flow manager has the most important function of all the communication processors. Among the tasks for which it is responsible are:

1. Routing packets to the appropriate output links according to packet destination, network condition and routing table information.

2. Periodically updating the routing-tables.
3. Detecting network component failures.
4. Recovering from component failures.
5. Controlling the message input rates to avoid overload.
6. Detecting and controlling errors.
7. Acknowledging packet receipt.
8. Collecting network communication statistics.

Transmission Facilities

Transmission Signalling

There are two basic methods by which digital signals are transmitted. They can be transmitted as they are, without modulation, or they can be superimposed or modulated upon a higher frequency which carries them. Without modulation, a signal cannot be transmitted over long distances due to signal distortion caused by the transmission line. If the signal is to be modulated, then it can either be transmitted in an analog or digital form.

Baseband Transmission. The transmission of data signals at their original frequency and shape is called baseband transmission. Baseband transmissions can be transmitted over open wire pairs for distances of a few miles and at speeds up to 300 bits per second. (Ref 14: 119-128).

Analog Transmission. Nearly all existing computer networks rely on commercial telephone facilities for data transmission. Today's commercial telephone facilities are designed for voice transmission and almost all such facilities use analog transmission with frequency division multiplexing. With proper modulation techniques, up to 4800 bits per second can be derived from a dial-up line and with proper line conditioning up to 9600 bits per second can be derived from a voice-grade leased line (Ref 13:1314).

Digital Transmission. In constrast to analog transmission, digital transmission uses a train of high rate pulses to carry the transmitted information instead of a sinusoidal or analog carrier. Using pulse code modulation techniques, transmission rates of up to 56,000 bits per second are possible using the Bell System's Tl carrier (Ref 14:119-128).

Terminal Access and Trunk Transmission Lines

Terminal Access Transmission Lines. Terminal access transmission lines are the transmission lines on the slower speed side of the multiplexers, concentrators, communication processors, etc., connecting the various terminals to the network. With the exception of remote-job-entry and graphics terminals, terminal transmission rates are usually lower than 2400 bits per second. The remote-job-entry and graphics terminals transmission rates are usually no higher than 4800 bits per second. Consequently, a terminal access transmission line is usually a subvoice line, a voice-grade line with a medium speed moden, or a dial-up connection.

Trunk Transmission Lines. Trunk transmission lines are the transmission lines on the higher speed side of the multiplexers, concentrators, communication processors, etc., connecting the various communication processors to one another or to the host computers. The type and rate of the transmission lines used as trunk lines tends to vary with the global topology of the computer network. In a ring network, very high speed 1,544 megabits per second lines are usually used. In a centralized network, the lines merging from a multiplexer or a data concentrator usually range in the transmission rate from 2400 to 9600 bits per second. In store-and-forward packet-switched distributed networks, the trunk line transmission rates rang upwards from 9600 bits per second.

Transmission Management Software/Hardware

In a computer network, it is essential that a series of basic transmission ccntrol procedures be established to ensure the efficient and accurate transfer of information within the network. The primary functions of these control procedures are:

1. To insure efficient and orderly use of transmission resources.
2. To make the network convenient to the users.
3. To prevent the loss of information.
4. To perform error detection and correction.
5. To detect and recover from network component failures.

6. To prevent and recover from transmission overloads and deadlocks.

These procedures differ in their degree of complexity as they are applied to the three different global network topologies. However, they can generally be partitioned into three categories:

1. Communications Protocols.

2. Flow control strategies.

3. Routing strategies.

Communication Protocols

A communication protocol is a series of procedures established to manage the exchange of information between two network communication components. The protocol provides standard format and code representations which allow the communicating components to understand each other.

Communication protocols can be classified into five levels depending on the network components with which they are associated (Ref 15:271-279). A given computer network may not use or require all of the five protocol levels.

Line Control Procedures. This is the lowest level communication protocol. It manages the physical transmission medium and does some automatic error detection and correction.

Communication Processor Pair Procedures. This protocol level provides transmission error detection and correction, flow control, and routing.

Host Computer/Terminal and Communication Processor Pair Procedures. This protocol level establishes the procedures which permit a host computer/terminal to transmit messages to other host computers/terminals and have the message's receipt acknowledged.

Host Computer Pair Procedures. This protocol level allows a number of host computers to transfer information between processes running on separate computers.

User Process Pair Procedures. This is the fifth and highest level of communication protocols. It provides user processes (e.g. an interactive terminal) with a general set of language primitives which isolate them from many of the details of the operating system and the transmission operation and control procedures.

Flow Control Strategies

A flow control strategy is a set of procedures which regulate the acceptance of information for transfer into the network. Flow control procedures regulare the amount and rate of information a communication processor can accept. This is done in order to prevent or minimize the occurrence of transmission overloads and deadlocks. As with communication protocols, flow control strategies differ in their approach and complexity with the global topology of the computer network. However, they all basically achieve the required control of transmission flow by allowing a message to be input only if a buffer has been reserved for it, and/or by limiting a message input if the number of occupied buffers reaches a previously specified limit (Ref 16:422-515).

Routing Strategies

A routine strategy manages the output queues in the various communication processors. It decides when and where a message should be transmitted. Its objective is to minimize message delay time and at the same time optimize the message throughput. The routing problem in networks with a centralized or ring global topology is normally quite simple: for there is usually only one transmission path available from origin to destination. In such cases, a message must simply wait until the path it must use is not busy in order to be transmitted.

For the distributed global topology network, however, the routing problem is far more complicated. Routing strategies for such networks do exist and can be characterized as being a deterministic or adaptable strategy, as providing centralized or distributed control, and as utilizing single or multiple transmission paths (Ref 16: 270-421).

Deterministic and Adaptive Strategies. In a deterministic routing strategy, message packets being transmitted between a given pair of communication processors will always be routed over the same transmission path, unless there is a network component failure. In an adaptive routing strategy, message packets being transmitted between a given pair of communication processors will not necessarily always be routed over the same transmission path. The path used will be determined dynamically, based upon network conditions at the time the transmission request is made.

Centralized and Distributed Control. Centralized routing control calls for all route determination to be made by a single communication processor. Distributed routing control calls for a best route to be determined by each originating communication processor.

Single and Multiple Paths. As its name implies, a single path routing strategy provides for only one transmission path between a given pair of communication processors. A multiple path strategy provides for two or more distinct transmission paths between a given pair of communication processors.

Summary

There are basically three types of computer networks; ring, centralized, and distributed store-and-forward. Each type of network is composed of three primary sets of components; communication devices, transmission facilities, and transmission management software/hardware. Communication devices carry out the functions of message switching, network control, and interfacing. Transmission facilities interconnect communication devices, terminals, and host computers. Transmission management software/

hardware ensures the efficient and orderly transfer of information between the various components of a computer network.

The parts which make up the whole of a computer network are numerous and varied. However, a basic awareness and understanding of these parts is essential in any effort to analyze and model a computer network.

IV. COMPUTER NETWORK PERFORMANCE MEASURES

Introduction

A basic understanding of computer network performance measurement is essential throughout the computer network analysis cycle, i.e., from problem definition to simulation experimentation. However, the measurement of computer network performance is difficult and at times totally qualitative in nature. This is due to the multitude of different components (hardware and software) that are to be found in a computer network. To evaluate a network effectively, a set of performance measures that encompasses the network topologies, communication devices, transmission facilities, and transmission management software/hardware and treats them as a single system, must be employed.

The National Bureau of Standards (NBS) (Ref 17) defines nine measures for evaluating computer network system performance. They are:

1. Availability.
2. Reliability.
3. Transfer Rate.
4. Accuracy.
5. Channel-Establishment Time.
6. Network Delay.
7. Channel-Turnaround Delay.
8. Transparency.

9. Network Security.

These nine measures do not represent all the possible performance measures, but they are considered to be the most essential and can be applied to any computer network and provide a basis for comparison with other networks.

Availability

Availability is defined as that portion of a specified time interval during which the transmission path is capable of performing its function. Availability is expressed as a percentage.

This definition is commonly used when discussing data transmission systems. However, it should be noted that American Telephone and Telegraph (AT&T) defines availability less rigorously as the percentage of time during which a telephone channel is operating within its specifications. In comparing the first (NBS) definition with the AT&T definitions, it should be noted that the second definition includes only the transmission channel. This means that the percentage availability as defined by the NBS definition will be much lower than the percentage availability based on the AT&T definition.

A computer network's availability is decreased not only by component failures but also by transmission overloads caused by either messages contending for a transmission channel or by the inability of the transmission processing equipment to handle all the transmission requests made to it at a given instant of time. The network availability should at least fall within the range of the associated host computers availabilities.

Reliability

Reliability is defined as the probability that a transmission request has been successfully completed. Expressed as a percentage, reliability differs from availability in that it describes the performance of a network after it has accepted a transmission request from its source.

Directly related to network reliability is network survivability. While more of a concern in military computer network applications, overhead and not part of the useful information transferred. Obviously, the latter group will always arrive at a lower transfer rate.

Accuracy

Accuracy or residual-error rate or undetected-error rate is defined as the ratio of undetected error bits received at a terminal station to the number of information bits transmitted to that terminal station. The value of the residual-error-rate (RER) is computed by the equation

$$RER = (C_e + C_u + C_d) / C_t$$

Where

C_e = the number of erroneous information bits accepted by the receiving terminal station.

C_u = the number of information bits transmitted, but not received by the terminal station.

C_d = the number of duplicate information bits accepted by the receiving terminal station, though they were not intended for duplication.

C_t = the number of information bits in the total transmission.

Accuracy is expressed in the number of error bits per the number of information bits transmitted.

Channel-Establishment Time

Channel-establishment time is defined as the period of time required for network communication devices, transmission facilities, and transmission management software/ hardware to connect a calling terminal station with a called terminal station. The channel-establishment time includes both the time required to place the transmission request and the time required for the network to complete the connection. Channel-establishment time is normally expressed in terms of seconds.

The significance of a channel-establishment time varies with the network user applications. If a user expects to have lengthy dialogs survivability can be defined in the same manner as is reliability. The difference in the two terms lies not in their basic definition but in the ways in which a network might experience component failures. In considering reliability, internal network component failures are of primary concern. In considering survivability, external network component failures such as those caused by hostile attacks, acts of nature, etc., are of primary concern.

The approach most often taken in overcoming possible reliability and/or survivability problems is simple redundancy, i.e., providing alternate transmission facilities and communication devices. Additionally, the transmission management software/hardware can provide component failure detection and automatic component switchover to minimize the mean-time-to-repair.

Transfer Rate

Transfer rate is defined as the ratio of the number of information bits accepted by a receiving terminal during a single information transfer period to the duration of the transfer period. Transfer rate is expressed in bits per second.

In calculating network transfer rate, the transmission channel capacity specified by the carrier is the upper limit for the transfer rate. In an operating network, the transmission management procedures, propagation delay, channel turnaround time, and the retransmission of erroneous messages all subtract from this transfer rate upper limit. The point should be made that there are differing definitions of just what constitutes an information transfer. Most communications engineers consider transmission block headers and trailers to be part of the useful information transferred. On the other hand, some applications-oriented users consider such headers and trailers as simply with a number of the network members, he may find channel-establishment time to be quite significant.

Network Delay

Network delay or message transfer time is defined as the period of time required for a message to be transmitted from a source terminal station to a sink terminal station. Network delay is usually recorded in terms of milliseconds.

The network delay depends on the characteristics of the terminal facilities, i.e., their buffer capacities, the volume of information being transferred, the protocol handshaking required, the mode of transmission channel operation (half-or full-duplex), and the reliability of the transmission channel itself.

Long network delays in an interactive environment involving short bursts of conversational infomration exchange are undesirable. This is because an operator's typing speed and/or the data input process would have to be slowed down and thus become inconvenient and annoying to the user. Long network delays, especially on the half-duplex transmission channels, seriously reduce the information transfer rate for short blocks of information by greatly increasing the time required for block acknowledgements. Such delays tend to force the use of a longer information block and greater terminal buffer storage.

Channel-Turnaround Delay

Channel-turnaround delay is defined as the time required by a half-duplex transmission channel to reverse its direction of transmission. Full-duplex channels do not incur turnaround delays because transmission in each direction is permanently established.

Channel-turnaround delay reduces the information transfer rate for half-duplex channels by increasing the time required for block acknowledgements. As with network delay, this delay can be offset to some extent by using longer message blocks and larger storage buffers.

Transparency

Transparency is a totally qualitative term which describes the lack of code or procedural constraints imposed on network information processing by network communication devices, transmission facilities, and transmission management software/hardware.

The lack of transparency bespeaks a need for changes in the network components or a need for user education in order to avoid conflict between the information processing and the communication portions of the computer network.

Most transmission protocols have a problem with code transparency. For example, in the basic mode control procedure (ANSI X3.28-1971) an end-of-transmission code cannot be used within the text of a message, because it may cause a channel disconnection. As a result, the entire message text must be scanned and modified in order to remove any illegal bit sequences. This detection and modification process, which may be handled in either hardware or software, extends the length of the message, reducing the information transfer rate. In a worst-case situation, the message actually transmitted thru the network might be nearly twice as long as the originally specified message. Transparent protocols do, however, exist. Digital Equipment Corporations' DDCMP Protocol is an example.

Network Security

Network security is another totally qualitative term used to describe the degree of protection afforded information handled in a computer network from unauthorized access. The importance of network security, of course, depends on the sensitivity of the information being handled. For most military and governmental networks, security is of considerable importance. For many commercial networks, the threst of industrial espionage and malicious damage place importance on network security.

The security of information is effected by the host computers, the communication devices, the transmission facilities, the transmission management software/hardware, the terminal station, the terminal station user authorization to use the network, and the isolation of user's files from each other. Of all of these network components, the communication devices, the transmission facilities, and the transmission management software/hardware are the most vulnerable to the unauthorized access of information.

Distributed networks provide an inherent but limited degree of security due to the alternative routing of successive message packets. This makes eavesdropping on any single channel less effective in that it may not provide access to the complete message. A distributed network can also set up a closed users group. This is a logically defined network of users within the physical network. Access to this logical network by unauthorized network members is prevented by the transmission management software/hardware. The most effective method of providing information security, however, is through information encryption. Encryption can be on an end-to-end or a channel-by-channel basis. The end-to-end encryption is the most effective in that it provides protection over the full length of the message's transmission path.

Summary

An awareness and knowledge of computer network performance measurement is a fundamental ingredient in any computer network analysis and/or design effort. The National Bureau of Standards (Ref 17) has defined a series of nine essential criteria for measuring computer network system performance. These nine criteria do not represent all the possible performance measures. For example, a computer network manager might be interested in the utilization rates of the major network components, the mean job throughput time, the various network queue lengths, the mean transmission delay times and where they occurred, etc.

All of the nine major and as many as is felt appropriate of the numerous minor network performance measures should be considered throughout the acquisition process, from network specification to network operation. The degree and manner in which these measures can be considered, of course, varies during the process of going from a conceptual design to an actual implementation in hardware and software. This variance is due to the fact that some of the performance measures, such as reliability and availability can be applied to a network while it is still in the design phase with a greater degree of accuracy than some of the other performance measures, which require an actual selection of hardware or software before they can be accurately considered. With regard to the numerous minor performance measures that can be considered, their use is really dependent on the desires and needs of the network analyst and/or designer.

V. SYSTEM MODEL DESCRIPTION

Introduction

This chapter describes the general computer network system model that was generated in order to satisfy the analysis objectives specified in Chapter I. The system model resources are described first in terms of the components which compose the model network, i.e., the host computers, the communication devices, the transmission facilities, and the transmission management software/hardware. Next, the system model's basic network configuration cost analysis is described. Then the system workload is defined and its flow through the network system is traced. Finally, the system performance measurements that were actually used are defined. In all cases the descriptions include the mathematical and logical relations that define the interrelationships of system entities, attributes, and subsystems. Flowcharts of the entire computer network system model are shown in Appendix A.

System Model Resources

Host Computers. The sole information processing devices within the network are the nodal processors. The characteristice (control parameters) that are used to describe each processor are processor type, processor location, processor main memory, number of control points, maximum input queue length, number of input/output channels, and processor work size. A discussion of each characteristic and its significance follows:

1. Processor Nomenclature. This characteristic is simply provided as a means of identifying, by the analyst, the nomenclature of the processor located at the given site or network node.

2. Processor Location. This characteristic is simply informational in nature and provides the analyst with a means of recording the geographical location of the processor.

3. Processor Main Memory. This characteristic specifies the amount of main memory available to the processor in kilo-words (only multiples of 1024 words are allowed). The memory management technique used is one in which "garbage collection" is performed as soon as the new block of memory becomes available (Ref 18:471-514). Thus, the currently available main memory space is always in a contiguous area. If a job will not fit into the available space, it will be placed into a memory queue until enough space is available.

4. Number of Control Points. This characteristic specifies the degree of multiprogramming possible for the processor. That is, the maximum number of jobs that can simultaneously be in main memory and thus be eligible to use the central processing

unit (CPU) (Ref 19:22). A control point can be viewed as a small communications area set aside in main memory. Each control point stores information unique to a particular job, such as, job identification number and priority that is required for the CPU to begin execution of a job. The number of control points is an integer variable.

5. Maximum Input Queue Length. This characteristic is an indirect specification of the amount of secondary mass storage, i.e., high speed disk, drum, or magnetic tape, that is available to the processor for the purpose of spooling incoming jobs. The queue length is written in terms of numbers of jobs. The method used to calculate the maximum queue length is as follows:

$$\mathrm{MIQL} = \frac{\mathrm{MJL}}{\mathrm{ASMS}}$$

Where:

MIQL - The maximum input queue length in jobs for a given processor.

MJL - The mean job length, in kilo-words, for a given processor.

ASMS - The total amount of secondary mass storage available for the spooling of incoming jobs, in kilo-words.

The maximum input queue length is an integer variable. If, following calculation, the MIQL is not an integer, it should be rounded up to the next higher integer.

6. Number of Input/Output (I/O) Channels. This characteristic is specified as a constraint on the handling of I/O requests. That is, a given processor possesses a limited number of I/O channels and the I/O requests of the individual jobs, being processed by the processor, must contend for their use. A given I/O channel, once it has been assigned for use by a particular I/O request, is not available for use by another I/O request. If an I/O channel is not available to service a particular I/O request, that request is placed into an I/O queue where it waits until a channel becomes available. The I/O channels described here are not to be confused with the transmission links between host processors.

7. Processor Word Size. This characteristic, while primarily informational in nature, is provided as an aid in the calculation of mean transmission time which will be discussed later.

Communications Devices. The presence of communication devices, i.e., multiplexers, data concentrators, front-end processors, modems, etc., is not explicitly represented in the system model. This is because it was felt that to include an explicit description of such communication devices added detail and complexity to the system model which was unnecessary for the fulfillment of the analysis objectives stated in Chapter I. However, the existence and some effect of a communication device can be represented in the form of a delay time which is used in the calculation of mean transmission time. Consequently, all processing of transmission requests between host processors is performed by the host processors themselves.

Transmission Facilities. The system model's transmission facilities are characterized as analog and/or digital trunk transmission lines. That is, the transmission facilities described are limited to the transmission links interconnecting the various host processors. The characteristics used to describe each transmission link are its terminal connections, its bandwidth, its length, its mileage charge, its terminal service charge, its transmission mode, and its mean transmission time. A discussion of each characteristics and its significance follows:

1. Terminal Connection. This characteristic is used to specify the terminal (processor-to-processor) connections of a given transmission link. Additionally, the terminal connections as a set specify the global network topology, i.e., ring, centralized, or distributed. A particular link's terminal connnections are described by an unordered tuple (P_i, P_j).
2. Bandwidth. This characteristic indicates the capacity of the transmission link, i.e., the number of information bits that can be transmitted along the link in a specified unit of

time. Bandwidth is a real variable and is expressed in kilo-bits per second (KBS).

3. Length. This characteristic states the physical length of the transmission link. Length is a real variable and is stated in miles.

4. Mileage Charge (Ref 20:577-582). This characteristic provides the analyst with the cost per mile per month of the transmission facility. This cost is of special significance when describing a network which uses commercially provided transmission facilities. More specifically, it refers to the cost of that part of the transmission facility located between commercial exchanges. This cost will be used in calculating the overall network cost and is stated in terms of dollars/mile/month.

5. Terminal Service Charge (Ref 20:577-582). This characteristic provides the analyst with the cost per month for the termination services of the transmission facility. As with the mileage charge, this cost is of special significance when describing commercially provided transmission facilities. Specifically, it provides a cost for that part of the transmission facility between the local exchange and the local host processor. This charge is a fixed monthly charge. As with mileage charge, the terminal service is used in calculating overall network cost. Terminal service

charges are given in dollars/month.

6. Transmission Mode. This characteristic indicates the mode of information transmission over the transmission facility. In the system model a message-switched half-duplex transmission mode was chosen because message-switched networks are more likely to be used in applications where there is not a large volume of traffic between geographically widespread users (i.e., an Air Force Base) (Ref 14: 119-128).

7. Mean Transmission Time. This characteristic is defined to be the mean time that is required for the transmission of a message and the receipt of the transmission acknowledgement (Ref 21:1-54). This characteristic is in actuality the most significant of the various transmission facility characteristics and plays the biggest role in the system model. This is because it incorporates a number of the previously discussed characteristics and a number of previously undiscussed characteristics into the one characteristic of mean transmission time. It can be considered as the message transfer time or network delay time associated with a particular transmission facility (See Chapter IV). Mean transmission time is calculated as follows:

$$MTT = \frac{MML}{BW} + D$$

or

$$MTT = MML + 2d_{tp} + d_{ctd} + d_{cet} + d_{ack}$$

Where:

MTT - The mean transmission time in seconds.

BW - The bandwidth of the particular transmission facility in bits per second.

MML - The mean message length in bits (equal to mean job length for network X processor word size).

D - Additional delay incurred in secs. D is equal to $2d_{tp} + d_{ctd} + d_{cet} + d_{ack}$.

d_{tp} - The delay due to the physical transmission path. That is the length of the transmission facility and the internal delay in the various communication devices. That figure expressed in seconds, allows for a round-trip delay (i.e., send message and receive acknowledgement).

d_{ctd} - The channel-turnaround delay, expressed in milliseconds, required by a particular half-duplex channel to reverse its direction of transmission. It is incurred thru the necessity of having a message receipt acknowledgement.

d_{cet} - The delay in seconds due to channel-establishment time. That is, the time required for a transmission path between the communicating processors to be established. This time is considered only in the case of message transmission from source to terminal. The transmission path is considered to be established for the acknowledgment.

d_{ack} - The delay in millisecs in formulating the message receipt acknowledgement.

The mean transmission is expressed in seconds. It is associated with each individual transmission link or terminal connection tuple.

Network Cost Considerations

The system model considers the costs of various network software and hardware components only in a most rudimentary fashion. The intention is to provide the analyst with a rough idea of the costs associated with a particular computer network configuration. Thus, allowing a basic cost comparison between alternative configurations.

Four costs are incorporated in the system model's calculation of overall network costs. They are:

1. Hardware rental and maintenance cost/month.
2. Software rental and maintenance cost/month.
3. Transmission link cost/month.
4. Total cost of network configuration/month.

Hardware Costs. A monthly hardware rental and maintenance cost is associated with each processor site in the network. The cost figure itself is considered to be the sum of all rental and maintenance charges pertaining to the site's host processor, I/O devices, interactive terminals, etc., that the analyst may envision being located at the particular host processor site.

Software Costs. A monthly software rental and maintenance cost is calculated for each host processor site in a manner similar to the monthly hardware costs. An exception exists with regard to the costing of the network's transmission management software. The cost of this software is divided among the individual host processor sites.

Transmission Link Costs. A monthly transmission link cost is calculated for each transmission link or terminal connection tuple. The enumeration and calculation of this cost is depicted by the system model in greater detail than those associated with host processor hardware and software because the real difference between configuration costs is found in the transmission link cost. The transmission link cost is a sum of the individual transmission link costs. The individual costs are calculated as follows:

transmission link cost = (length x mileage charge +
terminal service charge.

Total Network Cost. A total monthly network configuration cost is provided to the analyst. This configuration cost is the simple sum of the host processor hardware, the host processor software, and the transmission link monthly costs.

System Workload Characterization

The workload characterization of any system must be made in accordance with the objectives of the system analysis. If the primary objective is to analyze system effectiveness, as is the primary objective of this analysis, the workload characterization would be different from that used in an analysis of system efficience. The major difference being that an effectiveness study considers the system user to a much greater extent than does an efficiency study.

The effectiveness of a particular computer network configuration can be discussed only in the context of what the computer is required to do (Ref 19:10-12). The network user's applications, when translated into program processing and data base access requirements, can be characterized by the type and the amount of system resources the network will have to provide in order to fulfill these requirements. The sum of system resource requirements generated by all the network users represents the system or network workload. Generally, the workload of a computer system has certain basic statistical characteristics that do not change radically over reasonably long periods of time. The use of such characteristics make it possible to (Ref 19:10):

1. Characterize the system workload by statistical distributions of requirements placed on individual system resources.
2. Define a unit of work and then express the workload characteristics in relation to this unit.

The workload characterization of the general computer network system model is based around a batch input environment. Simply, programs or jobs are input at a particular host processor and use system resources at that host processor. If processing is required at another host processor in the network, the jobs are transmitted (using system resources) to that host processor where they again use system resources. This process is continued until all the jobs' required processing is completed and the jobs are output at the originating host processors. Specifically, the network workload is characterized in terms of each host processor inputting a series of job types. Each job type requires a certain amount of processing or file requests to be performed within the network (Ref 22). Each file request is in turn divided into a number of job segments which contend for processor resources (Ref 23:1431-1442). This section will first describe what constitutes a job type and then a job segment.

Job Type. The programs or jobs input at a given host processor are described as a series of job types. A job type is characterized according to its cumulative probability of occurrence, the number of network file requests it requires, and the location of the requested files within the network. For example, suppose there is a base-level logistics computer network with eight processing locations Table I shows ten possible job types and their hypothetical probability of occurrence for jobs input at processor location 3.

Job type 1 describes those programs which require only one network file request. That file request is to be made at the originating host processor (host processor 3) itself. The probability of a job in the job input stream of host processor 3 being of this type is 65 percent. Job 2 describes those programs which require three network file requests; one of which must be made at processor 4. The probability of such a job occurring in the job input stream of processor 3 is 20 percent (0.85-0.65=0.20). Job type 10 describes those programs input at processor 3 which require access to files located at each of the network processor

TABLE I

JOB TYPE CHARACTERIZATION

JOB TYPE	NETWORK FILE REQUESTS NUMBER & LOCATION	CUMULATIVE PROBABILITY OF OCCURRENCE
1	3	0.65
2	3-4-3	0.85
3	3-1-4-3	0.89
4	3-2-1-4-3	0.92
5	3-5-7-1-3	0.93
6	3-8-6-5-4-3	0.95
7	3-2-6-4-6-3	0.95
8	3-1-3-4-1-6-5-3	0.98
9	3-8-6-4-2-1-5-3	0.99
10	3-1-2-4-5-6-7-8-3	1.00

locations. One percent of all jobs input at host processor 3 should be of this type.

Job Segment. Each file request is composed of a series of job segments. These job segments are the smallest units of work within the host processor which compete for host processor resources, i.e., main memory, CPU time, and I/O channel use. The job segments being processed at a particular host processor are characterized accordingly by the following descriptors:

1. The mean amount of main memory required per job segment (in kilo-words).
2. The mean number of I/O requests per job segment (a real variable).
3. The mean time per I/O request (in time units).
4. The mean CPU execution time between I/O requests (in time units).
5. The cumulative probability of a file request having N job segments, where N is from one to ten.

The first, second, and fifth descriptors are expressed in direct relation to a job segment. The cumulative probability of a file request having N job segments is expressed in the same manner as the cumulative probability of occurrence for a job type. For example, Table II shows the probability of occurrence of job segments for file requests at a host processor.

In the case of Table II, file requests processed at the particular host processor have a 20 percent probability of having just one job segment. They have a 60 percent (0.80-8.20=0.60) probability of having two jobs segments.

TABLE II

JOB SEGMENT CHARACTERIZATION

NUMBER OF SEGMENTS	CUMULATIVE PROBABILITY OF OCCURRENCE
1	0.20
2	0.80
3	0.85
4	0.90
5	0.95
6	1.00
7	0.0
8	0.0
9	0.0
10	0.0

No file request processed at this processor will have more than six job segments.

The third and fourth job segment descriptors are indirect in their relationship to a job segment. However, it is felt that their inclusion as job segment descriptors is justified through their direct relationship with the second job segment characteristic; the mean number of I/O requests per job segment.

Job Flow

A job's flow through the network system model is a flow through the queueing network shown in Figure 6. The flow is portrayed as a series of discrete events. The occurrence or timing of these events is on a next event scheduled basis and is governed according to the various statistical distributions of requirements being placed on individual system resources. This section will discuss job flow in terms of job type selection and processing, job segment selection and processing, I/O request processing, and inter-processor transmission processing.

Job Type Selection and Processing. The selection and processing of a job type involves its initial input into a host processor, the determination of its type, and the assignment of a job priority. The arrival of jobs at a host processor is a Poisson process with job inter-arrival times exponentially distributed (Ref 24:73-98) about a mean inter-arrival time. This distribution can be written as:

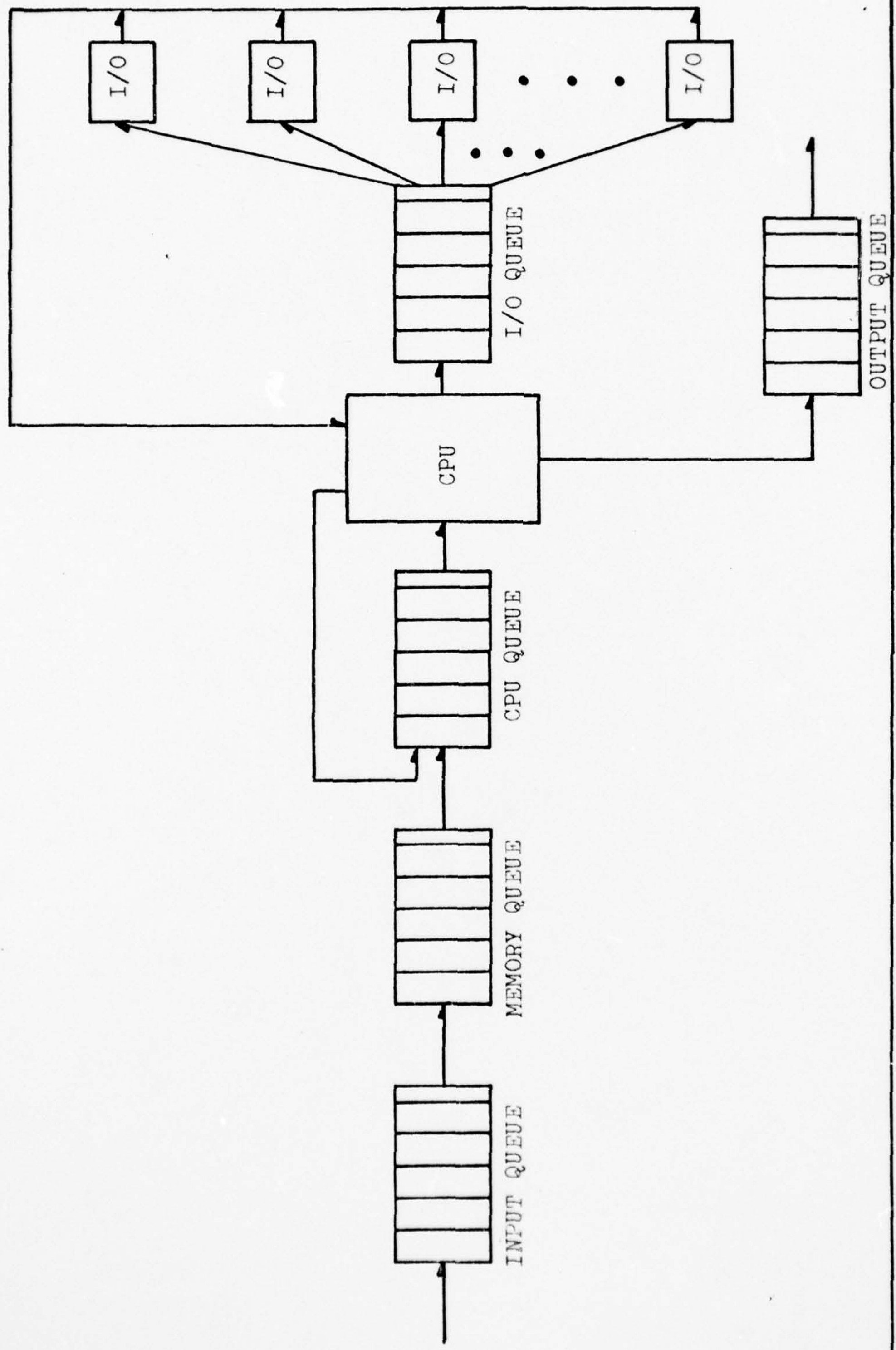


Fig 6. Queueing Network

$$P(t) = 1 - e^{-t/\lambda}$$

Where:

$P(t)$ - the probability that a job will arrive within time t.

λ - the mean job inter-arrival time at the particular host processor.

Since the jobs are assumed to enter the host processor's input queue independently and at random, P(t) must be a random number between 0 and 1. Equivalently, 1 - P(t) is a random number R. Thus,

$$P(t) = 1 - e^{-t/\lambda}$$

$$e^{-t/\lambda} = 1 - P(t)$$

Hence $$e^{-t/\lambda} = R$$

and $$-t(\lambda) = \ln R$$

or $$t = -\lambda \ln R$$

Where t is the time increment added to the present time to schedule the arrival of the next job at the host processor.

The type of the job input at a processor is determined according to what can be called a step probability function. This function can be described as follows:

1. Let R be a randomly generated number between 0 and 100;

2. Let P be the cumulative probability of job type 1 occurring;

3. Then if P is greater than or equal to R, let type equal 1. If P is less than R, let P be the cumulative probability of job type 2 occurring;

4. Continue this process until P is greater than or equal to R.

This step probability function allows for the independent random selection of job types while maintaining their relative probabilities of occurrence.

When a job is input at a processor it is assigned a job priority between one and ten. This priority remains with the job throughout its processing. That is, it does not change as it flows through the computer network system. The assignment of this priority is according to a step probability function. The basis for this function is an input statement of a priority's cumulative probability of occurrence at a particular processor, as shown in Table III.

In this instance, the likelihood of a job having a priority of 1 is 5 percent. The likelihood of it having a priority of 5 is 40 percent.

Job Segment Selection and Processing. The selection and processing of job segments begins after a job has been assigned a type and priority and has been placed into the appropriate host processor's input queue. The appropriate host processor input queue to be used is, of course, determined by the job type.

The input queue is a ranked set of queue based upon job priority. The job (file request) is placed into the input queue where it waits for a control point to become available. When a control point does become available, it is assigned to the file request with the highest priority. At this point, the number job segments contained in the file request and the memory required by the

TABLE III

JOB PRIORITY CHARACTERIZATION

PRIORITY NUMBER	CUMULATIVE PROBABILITY OF OCCURRENCE
1	0.05
2	0.10
3	0.15
4	0.45
5	0.85
6	0.95
7	0.97
8	0.98
9	0.99
10	1.00

first of these job segments is determined.

The number of job segments contained in the file request is defined by a step probability function using the cumulative probability of occurrence described earlier. The job segments thus generated all carry the same priority as the originating file request.

The amount of main memory required by the first job segment of this file request is assigned by a stuttering exponential distribution. That is, a regular exponential distribution with the mean memory required per job segment as lambda is used to generate a memory requirement. However, since memory calculations and descriptions are in terms of kilo-word blocks the memory requirement just generated is truncated (stuttered) to a kilo-word requirement. That is, the memory requirement is defined to be the largest kilo-word block less than that defined by the exponential distribution.

After a memory requirement has been given to the job segment, it is placed into a memory queue where it waits for sufficient main memory to become available. The memory queue is a ranked set based upon job priority. If sufficient memory is available for the highest priority job segment, the memory space is allocated to it and it begins CPU execution. If sufficient memory is not available for the highest priority job segment in the memory queue, the job segment remains in the queue and no further allocation of memory takes place, even if a lesser priority job segment could fit into main memory.

Once a job segment has been allocated space in main memory and begins execution, it retains control of the appropriate processor's CPU until it has either completed execution, i.e., has no I/O requests, or is interrupted by an I/O request. The amount of CPU execution time between I/O requests is determined by an exponential distribution with the mean CPU time between I/O requests as lambda.

All job segments wait while the highest priority job segment in main memory is executed. When this job segment completes its execution, the main memory space that was allocated to it is released along with the control point that managed its execution. The just released control point will be assigned to the next job segment, if one exists, of the file request whose job segment just completed execution. If there are no remaining job segments for this file request, the control point is assigned to a job segment of the next highest priority file request. The new job segment then enters the main memory queue and an attempt is made to fill the highest priority job segment now in the memory queue into main memory.

The memory fitting process continues until the highest priority job segment in the memory queue is able to fit into main memory. The job segment in main memory which has the highest priority executes in a host processor's CPU as if it were the only job segment in the system. It yields CPU execution time only when interrupted by an I/O request. During this time, the job segment in main memory with the next highest priority is allowed to execute.

I/O Request Processing. The number of I/O request interrupts that a particular job segment will experience during execution is defined by a stuttering exponential distribution with the mean number of I/O requests per job segment as lambda. The time interval between any two I/O requests is assigned by an exponential distribution with the mean CPU execution time between I/O requests as lambda. These distributions were discussed earlier. The duration of a particular I/O request is determined by yet another exponential distribution with the mean time per I/O request as lambda.

Once the job segment which was interrupted by an I/O request relinquishes control of the CPU it is placed into an I/O channel queue. The I/O channel queue is a ranked set based on high priority of job segments. If an I/O channel is available, the I/O request immediately begins its processing. If an I/O channel is not available, the I/O request remains in the I/O channel queue.

When an I/O request terminates, a series of four actions can take place. First, the I/O channel which was just released is assigned to the highest priority I/O request in the I/O channel queue. Second, a comparison of priorities is made between the job segment which is currently in control of the host processor's CPU and the job segment whose I/O request has just terminated processing. Third, if the job segment currently in control of the CPU is of a higher priority, it retains control of the CPU and continues execution. The job segment which has just completed its I/O request and is of a lower priority is placed into the CPU queue to await execution. The CPU queue is a ranked set based upon highest priority. When selecting job segments to receive CPU time, the CPU queue is addressed prior to the main memory queue. This is done because the job segments in the CPU queue are job segments which were previously selected from the main memory queue for execution and should be allowed to complete their execution prior to selecting a new job segment for execution. Fourth, and finally, if the job segment currently in control of the CPU is of a lower priority it is swapped out with the highest priority job segment that has just completed an I/O request. The lower priority job segment is then placed into the CPU queue.

Upon the completion of a file request, i.e., all the file request's job segments have been executed, it is determined if the just completed file request is the last file request of the given job type which requires processing. If it is the last file request, then the various job statistics are accumulated and the job is output at the originating host processor. If it is not the last file request, then the job and its associated accounting information must be transmitted to another host processor to begin processing of the next file request.

Inter-Processor Transmission Processing. The necessity for an inter-processor transfer of information originates with the job type description itself. That is, the job type description specifies the number of inter-processor transmissions that a particular job will make and the origin and destination (source and terminal) of each transmission. For example, job type 5 of Table I specifies that four inter-processor transmissions will be made with origin and destination pairs of (3,5), (5,7), (7,1), and (1,3) respectively.

A given inter-processor transmission is characterized according to three aspects. The first is the transmission's origin and destination pair. This information, as shown above, is specified by the job type description.

The second is the path taken by the transmission through the network from origin to destination. This transmission path is determined by the network's transmission management software/hardware. In the case of the network system model, path assignment is performed by a fixed-minimum-path routing algorithm.

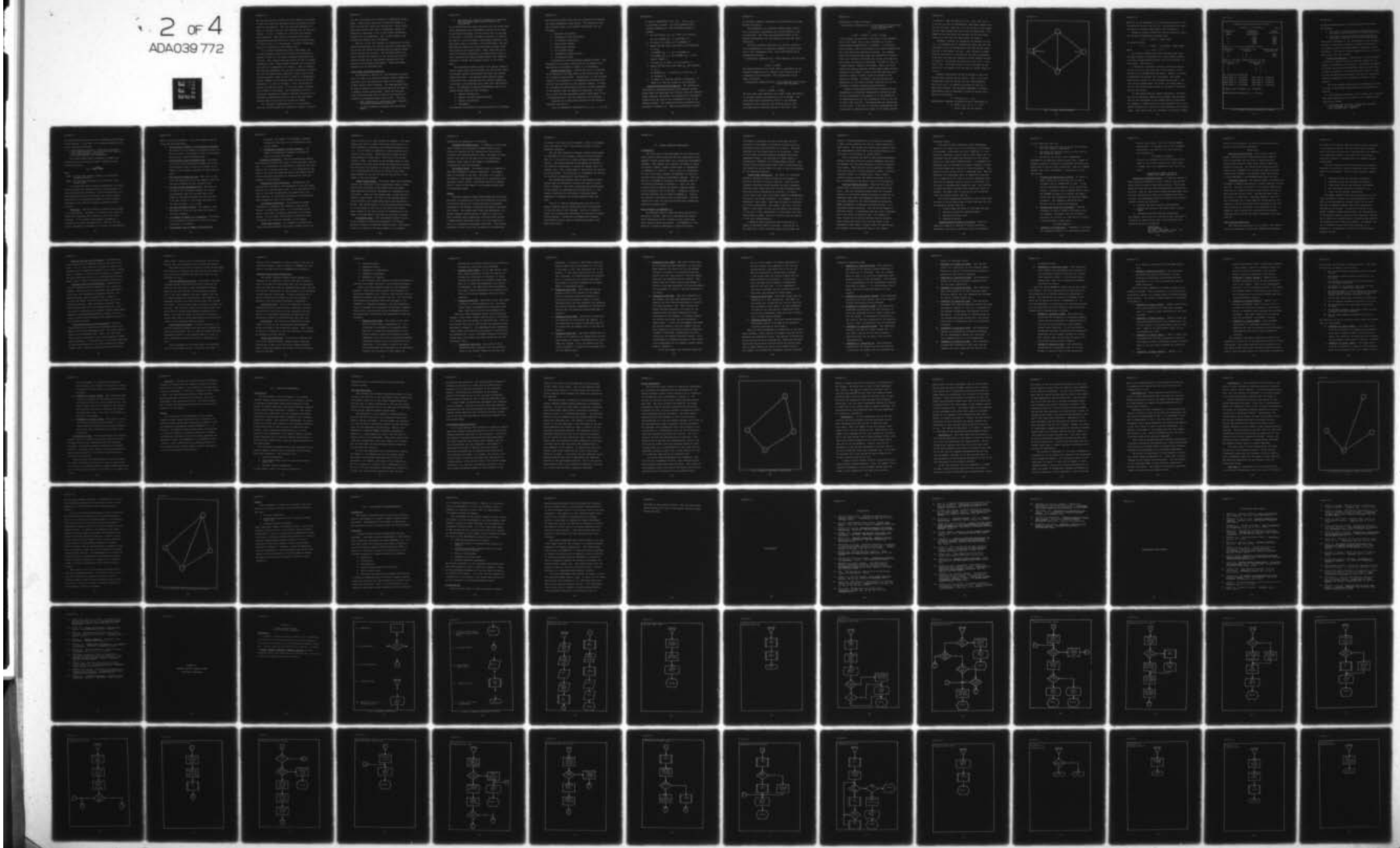
Ideally, a computer network's global topology should allow for a direct transmission link between all processor pairs, i.e., the network should be fully connected. However, due to economical and physical constraints, a fully connected network is not always feasible. The alternative is for the network to be partially connected. Herein lies the heart of the transmission routing problem. For, in a fully connected network, the choice of which transmission path to take is elementary. The only complication that can arise is that the direct transmission link may fail or become overloaded. On the other hand, in a partially connected network the choice of which transmission path to take is not as simple. Normally, there are a number of possible transmission paths between the origin and destination host processors. With each transmission path traversing several intermediate transmission links and host processors. The routing problem now becomes one of selecting the "best" transmission path from a large list of possible paths.

Different routing algorithms can be devised to determine the "best" transmission path between host processors. (See Chapter III). These algorithms can take into account a very large number of factors, such as, transmission time, network conditions, costs, etc., when selecting a transmission path. Of course, as the number of factors included in the path selection process grows so does the amount of system resources required to support the process. Consequently, the network system model uses a routing algorithm which selects a transmission path based upon the following two criteria:

1. The transmission path must traverse the least number of transmission links, from origin to destination, of all possible paths.
2. The number of transmission links traversed must not exceed a specified limit (in the network system model the limit is set at five).

AD-A039 772 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 9/2
A GENERAL COMPUTER NETWORK SIMULATION MODEL.(U)
MAR 77 H M WARREN
UNCLASSIFIED AFIT/GCS/EE/77-3 NL

2 OF 4
ADA039 772



The routing algorithm itself, initially checks to determine if both the origin and destination host processors are connected to the network. Then the algorithm determines if there is a direct (one-link) path between the two host processors. If not, it checks for a two-link path between host processors. This process continues until either a transmission path which does not exceed the maximum length is found or until it is determined a suitable transmission path is not available at the present time.

If a suitable transmission path is available, the routing procedure allows the job making the transmission request to seize all the system resources associated with the path. Thus, making them unavailable for use in another transmission path. These system resources will be unavailable until the transmission is complete. When a transmission has been completed, the job just transmitted is placed into the destination host processor's input queue and the routing procedure checks to see if there is a transmission request anywhere in the network which can use the system resources it is about to release. If so, the system resources are made available. If not, the system resources are released to the system. The transmission time of a particular transmission request is defined according to an exponential distribution with the sum of the mean transmission times of the links in the path as lambda.

If a suitable transmission path is not available, then the job which made the transmission request is placed into the originating host processor's transmission output queue. These output queues are ordered sets based on a first-in-first-out (FIFO) servicing policy. These output queues are checked each time a transmission request is completed to determine if any of the queued transmission requests can be transmitted. This is done before a new transmission request can be processed.

When all of the file requests of a particular job have completed processing, the various statistics associated with the job and the system are accumulated and updated respectively and the job is output. The statistics which are used to measure the behavior of the various jobs and the network system are the performance measures which are applied to the network system model. These performance measures will be discussed next.

System Model Performance Measures

As mentioned in Chapter IV, the performance measures which are made on an existing computer network depend predominately upon the desires and needs of the computer network analyst and/or designer. The same approach must be used in selecting the performance measures to apply to a computer network system model. There are basically three questions which must be asked in the selection of a specific performance measure (Ref 19:76). They are:

1. What information is required to meet a specific performance measurement objective?
2. Is this information available from the system model?

3. How should the required information be extracted from the system model and how should it be represented?

It is obvious from the above questions that the system model itself determines which performance measures can be made. As previously discussed, the more detail that is included in the model, the closer the model's depiction of reality. The same is true with regard to performance measures. The more detail that is included in the system model, the more detailed and numerous are the performance measures which can be made on the system model. Thus, there must be an iterative process of balancing the desire and/or need to make a specific performance measure against the resources required to include the necessary detail in the system model.

The performance measures used in the general computer network system model can be divided into two groups; those that measure system effectiveness and those that measure system efficiency. The measures which describe system effectiveness are concerned with the system's capability to process a given workload and to meet user time requirements. The measures of system effectiveness used in the network system model are the following:

1. Network connectivity.
2. Network reliability/survivability.
3. Network availability.
4. Throughput.

The measures which describe system efficiency are concerned with internal system delays and the utilization of individual system components versus demand. The measures of system efficiency used in the network system model are the following:

1. Processor utilization.
2. Transmission link utilization.
3. I/O channel utilization.
4. Input queue lengths.
5. Output queue lengths.
6. I/O requests delay.
7. Transmission request delays.
8. Job segment delays.

A discussion of each performance measure follows. This discussion will include a definition of the measure and a description of the manner in which it is calculated.

Network Connectivity. Connectivity as it is used here is not defined in the same sense as the connectivity used in a classical graph theory. Connectivity, in this instance, refers to the existence of a transmission path between all pairs of host processors in the computer network. If there is a transmission path between each pair of host processors in the network, then the network is connected. If there are at least two host processors between which there is not a transmission path, then the network is not connected.

Connectivity is determined according to the following algorithm (Ref 22:13):

Given N processors, represented as P_1, P_2, ..., P_n, and

M tuples, represented as (P_i, P_j). Let $S_1, S_2, \ldots,$ S_n represent n stacks. Let each processor have a pointer (POINTER(P_i)), and LET COUNT represent a counter.

1. For each couple (P_i, P_j), Place P_j on stack S_i and let POINTER (P_1) = 1 and COUNT = 1.
2. If stack S_1 is empty, go to statement 8.
3. Remove the top entry from stack S_1 and represent it as P_k.
4. If POINTER (P_k) = 1, go to statement 2. If POINTER (P_k) $\neq$ 1, set POINTER (P_k) = 1 and COUNT = COUNT + 1.
5. If stack S_{P_k} is empty, go to statement 2.
6. Remove the top entry from stack S_{P_k} and represent it as P_i.
7. If POINTER (P_i) = 0, place P_i on stack S_i, go to statement 5.
8. If COUNT = N, then the network is connected. If COUNT $\neq$ N, then the network is not connected.

Network Reliability/Survivability. The reliability/survivability calculations made by the computer network system model are based upon two factors; first the network's global topology; and second, the probability of successful communication for each of the transmission links in the computer network. The topologies which are possible were discussed in Chapter III. They are specified by a series of unordered tuples as described in the discussion on transmission facilities.

The probability of successful communication (Pr(SC)) for an individual transmission link can be defined in one of three ways. The first and second definitions do not address transmission link survivability as does the third definition.

The first possible definition for (Pr(SC)) treats it as the probability of no physical component failures occurring in the transmission link. This relationship can be expressed as follows:

Pr(successful communication) = PR(no physical link failures)

or

Pr(SC) = Pr(NPF)

The second definition for Pr(SC) sets it equivalent to the compound probability of no physical link failures plus no transmission link overloads. This relationship can be shown as follows:

Pr(successful communications) = Pr(no physical link failures)
+ Pr(no link overloads)

or

Pr(SC) + Pr(NPF) + Pr(NO)

The third and final definition for Pr(SC) allows the analyst to consider network survivability if he so chooses. The third definition equivalencies Pr(SC) to the compound probability of no physical link failures plus no link overloads plus no external attacks on the link. This equivalency is shown as follows:

$$\begin{aligned}\Pr(\text{successful communication}) = {} & \Pr(\text{no physical link failures}) \\ & + \Pr(\text{no link overloads}) \\ & + \Pr(\text{no attacks})\end{aligned}$$

or

$$\Pr(SC) = \Pr(NPF) + \Pr(NO) + \Pr(NA)$$

If the analyst has available to him each of the probabilities on the right, then a total statement of a link's Pr(successful communication) is possible. It should be noted, however, that the availability of these probabilities will vary during the acquisition process. For example, during the design process where the evaluation of alternatives is of concern, the Pr(NO) will not be available and must either be omitted or estimated. The same is true, to a lesser degree, regardint the Pr(NA). While the Pr(SC) used by the general computer network system model has been limited to the Pr(NPF), either of the two remaining probabilities (Pr(NO) + Pr(NA)) could be included if the analyst so chooses. Of course, the more detailed the description of the Pr(SC) the more realistic will be the estimate of network reliability/survivability.

Network reliability/survivability is defined as the probability of successful communication between any pair of host processors in the network or as the probability of communication of all the transmission links in all the cut sets (Ref 25:61-65). Two definitions are appropriate at this point. A tie set is a directed transmission path from origin to destination as shown in the simple network in Figure 7. The tie sets are (1,4), (3,4), and (2,4,5). A cut set is a set of transmission links whose removal cuts all the tie sets. That is, a cut set severs all the transmission paths between the origin and the destination. The number of cut sets which are possible can be very large. There are 16 cut sets for the network in Figure 7.

Normally, interest is focused on the minimal cut set, i.e., the smallest set of transmission links such that the exclusion of any one transmission link from the set would prevent it from being a cut set. This interest is based upon the fact that in a nominal cut set more transmission links must fail than are required to cause network failure. In Figure 7, the minimal cut sets are (1,4), (2,3), (2,4), and (1,3,5). Note that (2,4,5) is not a minimal cut set since (2,4) is already a cut set and is a proper subset of (2,4,5).

Network unreliability $\overline{R}$ can be defined as the probability that at least one cut occurs. That is, all the transmission links of the cut set fail simultaneously. Let C_i,i=1,2,3,....N denote the minimal cut sets of a particular network. The preceding statement of network unreliability can be expressed in terms of network reliability R as follows:

$$R = Pr(C_1 . C_2 . C_3 ..... C_n)$$

Equivalently, network unreliability can be expressed as:

$$1 - R = Pr(\overline{C}_1 + \overline{C}_2 + \overline{C}_3 + ... + \overline{C}n$$

$$= Pr(\text{at least one cut occurs})$$

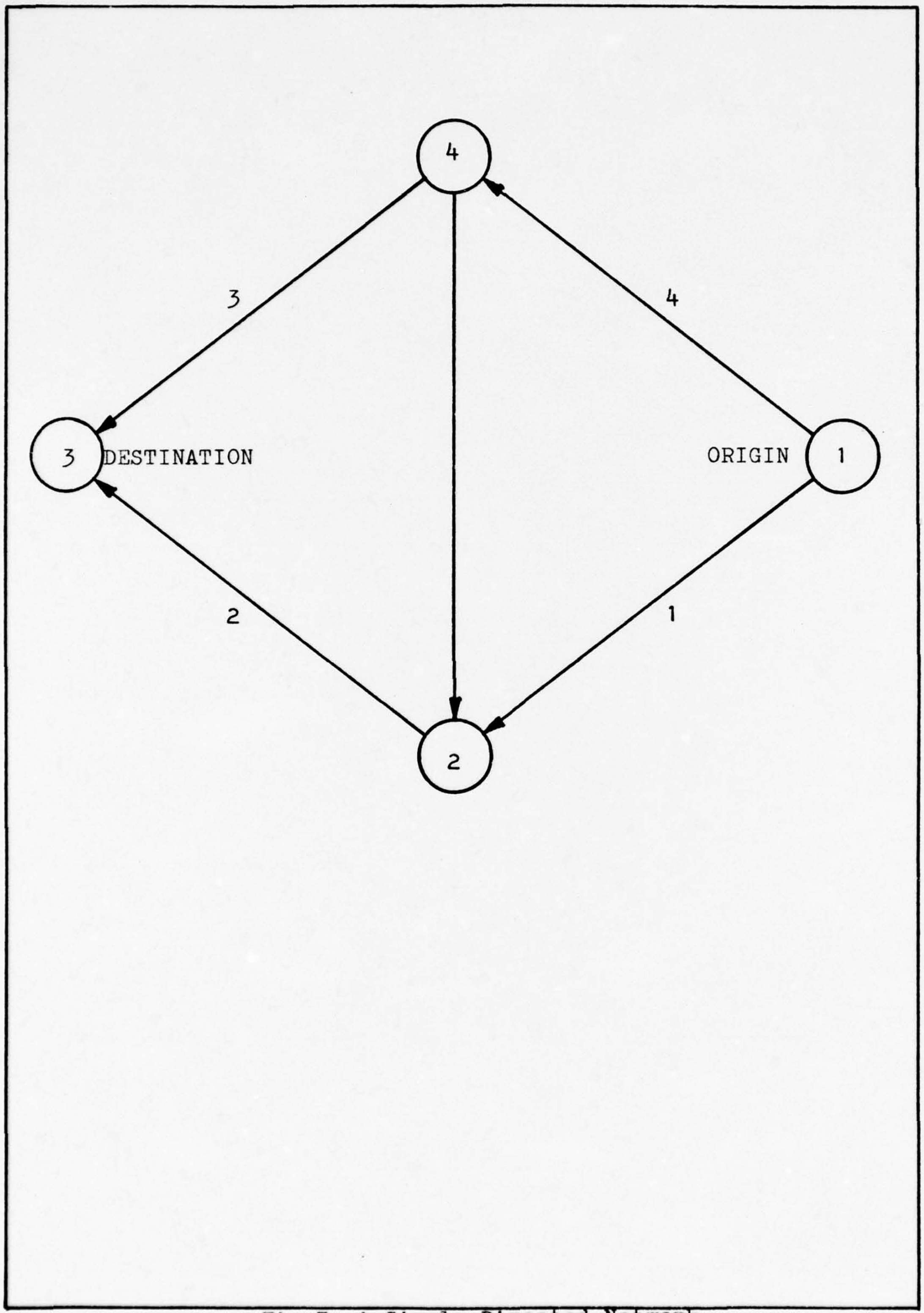


Fig 7. A Simple Directed Network

Where $\overline{C}_i$ is the complement of C_i and denotes failure of all transmission links of the ith cut. The preceding are exact equations for network unreliability and reliability.

Bounds on network reliability can be obtained by using the following probabilistic inequalities:

$$R \leq 1 - \Sigma \Pr(\overline{C}_i) = \text{Lower Bound}$$

by using two terms

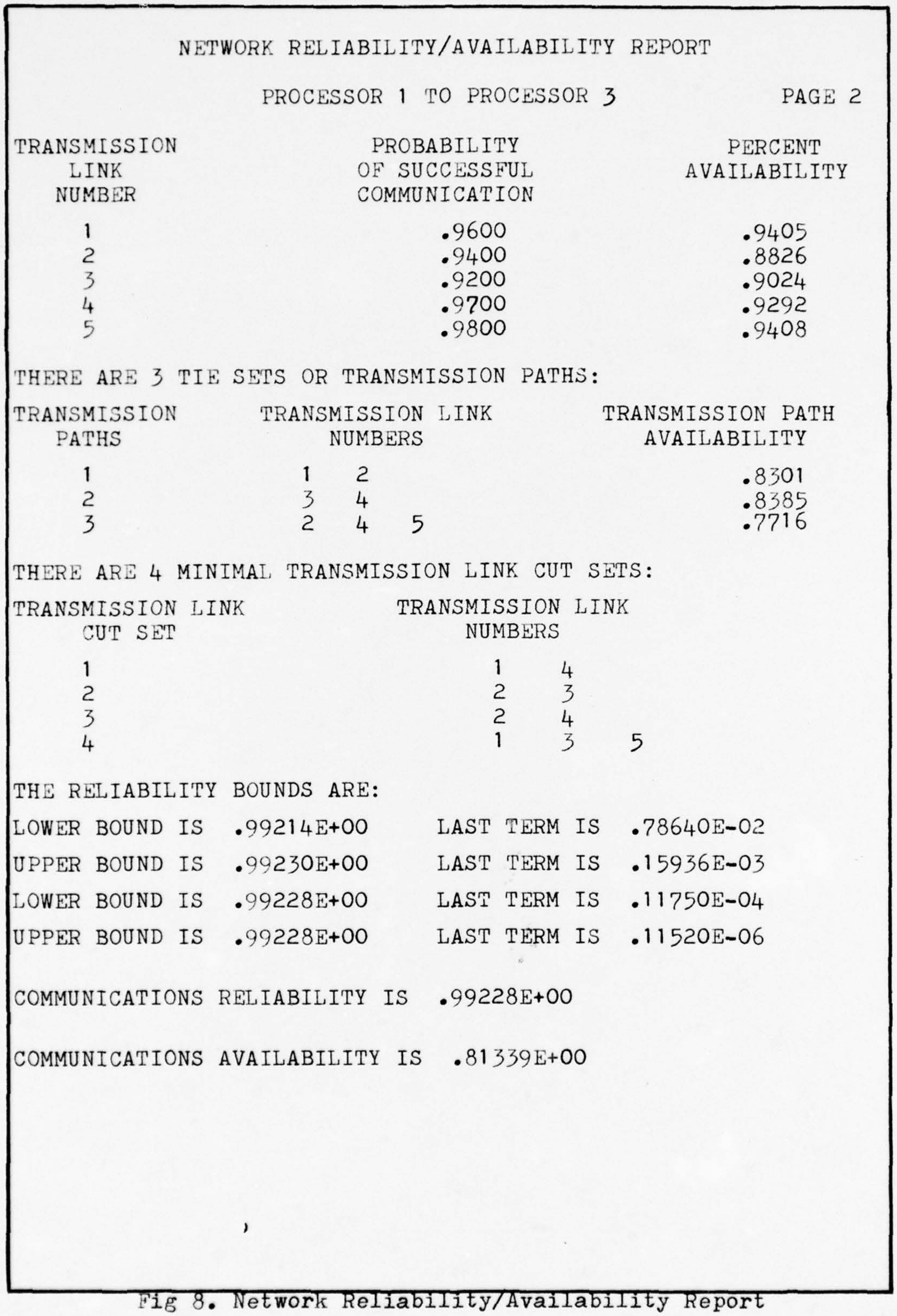
$$R \leq 1 - \Sigma \Pr(\overline{C}_i) + \sum_{i < m} \Pr(\overline{C}_i\overline{C}_m) = \text{Upper Bound}$$

Where $1 \leq 1$ and $m \leq n$

A process of adding terms and calculating new lower and upper bounds is performed until these bounds converge to within a predetermined epsilon (set by the analyst) of each other. See Figure 8.

A method for calculating network reliability/survivability using tie sets exist and is very similar to the minimal cut set method described above. However, as stated by Messinger and Shooman (Ref 26:292-304), the bounds based on the minimal cut set calculations are best in the case of high transmission link Pr(SC), and those based on the tie set calculations are best in the case of low transmission link Pr(SC).

In calculating overall network reliability, the probability of successful communication between each unordered pair of host processors in the network must be calculated. For example, a network with four host processors and a ring global topology will require six host processor pair calculations. Once these Pr(SC) are available, the overall network



```
               NETWORK RELIABILITY/AVAILABILITY REPORT

                    PROCESSOR 1 TO PROCESSOR 3              PAGE 2

TRANSMISSION             PROBABILITY                PERCENT
    LINK               OF SUCCESSFUL             AVAILABILITY
   NUMBER              COMMUNICATION

     1                      .9600                    .9405
     2                      .9400                    .8826
     3                      .9200                    .9024
     4                      .9700                    .9292
     5                      .9800                    .9408

THERE ARE 3 TIE SETS OR TRANSMISSION PATHS:

TRANSMISSION       TRANSMISSION LINK          TRANSMISSION PATH
   PATHS                NUMBERS                  AVAILABILITY

     1                1   2                          .8301
     2                3   4                          .8385
     3                2   4   5                      .7716

THERE ARE 4 MINIMAL TRANSMISSION LINK CUT SETS:

TRANSMISSION LINK              TRANSMISSION LINK
     CUT SET                        NUMBERS

     1                               1    4
     2                               2    3
     3                               2    4
     4                               1    3    5

THE RELIABILITY BOUNDS ARE:

LOWER BOUND IS  .99214E+00      LAST TERM IS  .78640E-02

UPPER BOUND IS  .99230E+00      LAST TERM IS  .15936E-03

LOWER BOUND IS  .99228E+00      LAST TERM IS  .11750E-04

UPPER BOUND IS  .99228E+00      LAST TERM IS  .11520E-06


COMMUNICATIONS RELIABILITY IS  .99228E+00


COMMUNICATIONS AVAILABILITY IS  .81339E+00
```

Fig 8. Network Reliability/Availability Report

reliability/survivability is calculated by the following two methods:

1. The overall network reliability/survivability is set equal to the lowest Pr(SC) of those calculated.

2. The overall network reliability/survivability if set equal to the mean of all Pr(SC)s calculated.

The primary difference in the two methods is that the first provides a pessimistic statement of network reliability/survivability while the second method's statement is more optimistic. The choice of which method to use is one which the analyst will have to make.

Network Availability. Availability A is the probability that at a specific point in time the network will be capable of performing its function. As with the calculation of network reliability/survivability there are three ways in which availability can be defined (Ref 27:9-11). Also, as with reliability/survivability, the use of a particular definition depends on the information available to the analyst.

The first possible definition for availability is the one used in the network system model and is expressed as follows:

$$A = \Pr(\text{No physical failures}) + \Pr(\text{i failures and i repairs})$$

The second definition for A treats it as a compound probability as follows:

$$\begin{aligned}A = {} & \Pr(\text{No physical failures}) + \Pr(\text{No path overloads}) \\ & + \Pr(\text{j overloads and j recoveries}) \\ & + \Pr(\text{i failures and i repairs})\end{aligned}$$

The third and final definition for A considers enemy attacks on the network. A once again is a compound probability and can be expressed as follows:

$$\begin{aligned} A = {} & \Pr(\text{No physical failures}) + \Pr(\text{No path overloads}) \\ & + \Pr(\text{No enemy attacks}) + \Pr(\text{i failures and i repairs}) \\ & + \Pr(\text{j overloads and j recoveries}) \\ & + \Pr(\text{k enemy attacks and k recoveries}) \end{aligned}$$

The network system model calculates a steady state availability for the overall network according to the following function.

$$A_{ss} = \frac{MTBF}{MTBF + MTTR}$$

Where

MTBF - The Mean Time Between Failure of a particular transmission path.

MTTR - The Mean Time To Repair of a particular transmission path.

In calculating overall network availability, the availability between each unordered pair of host processors in the network must be calculated. Once the availability for each transmission path is available, the overall network availability is calculated according to the two methods discussed in the section on network reliability/survivability.

Throughput. The last measure of network effectiveness is its throughput. Throughput is defined as the amount of useful work completed per interval of time with a given workload. In the network system model, throughput is measured at the host processor level. That is, the network's throughput is reflected in a series of eight measures made at each host processor. The eight measures used are listed and discussed below.

1. Job Priority vs Number of File Requests Processed. The number of file requests processed for each job priority level. This number is used in calculating Job Priority vs Mean Throughput Time.
2. Job Priority vs Mean Throughput Time. The ratio of the sum of the Differences between the arrival and departure times for each file request to the number of file request processes for each priority level.
3. Job Priority vs Mean CPU Time. The ratio of the total CPU execution time to the number of the file requests processed for each priority level.
4. Job Type vs Mean Throughput Time. The ratio of the sum of the differences between the arrival and departure times for each job type to the number of job types processed for each job type. A job type refers to a job input and output at the host processor, i.e., a job.
5. Job Type vs Number of Jobs. The number of jobs processed for each job type input and output at the host processor.
6. I/O Channel vs Number of I/O Requests. The number of I/O requests processed over each of the host processor's I/O channels
7. Transmission Link vs Number of Transmission

Requests. The number of transmission requests transmitted over each of the transmission links in the network.

8. Transmission Requests vs Host Processor. The number of transmission requests originated at each of the host processors.

Processor Utilization. The first performance measure associated with network efficiency is host processor utilization. It is defined as the percentage of time the host processor's CPU is doing useful work. It is calculated as the ratio of the total CPU execution time to the report period. A CPU utilization percentage is calculated for each host processor in the network.

Transmission Link Utilization. Transmission link utilization is defined as the percentage of time the transmission link is doing useful work. It is calculated as the ratio of the total transmission link usage time to the report period. A utilization percentage is calculated for each transmission link in the network.

I/O Channel Utilization. This measure of system efficiency is defined as the percent of time the I/O channel is doing useful work. It is calculated as the ratio of the total I/O channel usage time to the report period. An I/O channel utilization is figured for each I/O channel for each host processor in the network.

Input Queue Lengths. As a measure of network efficience the host processor input queue lengths reflect the flow or lack of it of jobs through the network. The input queues also tend to reflect the overall efficiency of a host processor in processing the workload presented to it, i.e., it can reflect an imbalance in host processor efficience in the network. Three measures are made on each host processor's input queue. The first is its current length in jobs. The second is the maximum length in jobs attained by the input queue during the observation period. The third and final measure is the mean input queue length. It is calculated as the ratio of the sum of the input queue lengths at the time of each new job arrival to the number of new job arrivals.

Output Queue Lengths. The output queue length measures reflect the efficiency of the transmission links in transmitting the job output by the network host processors. These measures can reflect imbalances in transmission link efficiency. There are two measures made on each host processor's output queue. The first is the maximum length in jobs attained by the output queue during the observation period. The second measure is the mean output queue length. It is calculated as the ratio of the sum of the output queue lengths at the time of each new transmission to the number of transmissions originated by each host processor.

I/O Request Delay. This performance measure is used as an indicator of each host processor's I/O channels ability to handle the workload presented them. The measure itself is in terms of the total number of I/O requests delayed due to unavailable I/O channels.

Transmission Request Delay. In addition to the output queue lengths, the transmission request delays can be used as an indicator of the transmission links ability to handle the workload placed on them. The delay is calculated as the ratio of the sum of all the transmission requests' delay times to the number of transmissions originated by each host processor.

Job Segment Delay. Job segment delay is a measure of each host processor's internal efficiency. Job segment delay is recorded in two ways. The first is a record of the number of job segments delayed due to unavailable host processor control points. The second records job segment delay in terms of the number of job segments delayed due to unavailable memory.

Summary

A general computer network system model has been built. The model is constructed to allow the evaluation of alternative network designs thru either modifying the workload presented to the network or modifying the network configuration itself. The system model allows the description of a general computer network design in terms of a series of host processors which communicate with each other through a communications subnetwork which is in turn managed by a series of transmission management algorithms. The workload presented to each of the host processors is characterized in terms of job types and job segments. With a job segment being the smallest unit of work which contends for host processor resources.

The job flow through the computer network system model is described as a flow through a series of network queues. That is, a job arrives at a system resource and demands service. The system responds by performing the service if the resource is available, or places the demand (job, I/O request, etc.) into a queue where it waits until the service can be performed. The occurrence of the demands for service are governed by probabilistic distributions.

The performance measures made on the computer network system model are designed to provide two forms of performance information. The first, describing the effectiveness of the network in processing the workload given it. The second, describing the efficiency of selected network components in handling the service demands placed upon them.

Finally, it must be emphasized that the use of the general computer network system model must be restricted to that for which it was intended. That is, to obtain a "feel" for the potential performance of alternative computer network designs. Extreme care accompanied by adequate qualification must be used in extending the model's use beyond this point!

VI. SYSTEM SIMULATION DESCRIPTION

Introduction

The fifth step in the development of the general computer network system simulation was the translation of the computer network system model into a simulation programming language. This chapter will describe the manner in which the general computer network system model is implemented in SIMSCRIPT I.5 (Ref 28). First, there is a discussion of some of the basic concepts of SIMSCRIPT I.5 used as a discrete event simulation programming language. This discussion is intended to reenforce and supplement some of the material presented in Chapter II. Next, the data structure used to support the general computer network system simulation will be defined. A complete listing of the data structure can be found in Appendix B. Finally, the simulation construction will be described. This discussion will cover the various sections which compose the simulation program itself.

Basic Concepts of SIMSCRIPT I.5

As discussed earlier, there are three objectives in analyzing a system. They are to understand how system state transitions occur, to predict state transitions, and to control these transitions. In SIMSCRIPT I.5, the state of a system is described in terms of Entities, Attributes of Entities, and Sets of Entities (Ref 29). The state of the System is changed at discrete points in simulated time by the occurrence of an Event. The Event describes how the System state is to transition. There are two types of Events possible; Exogenous Events and Endogenous Events. The occurrence of these Events is governed by a SIMSCRIPT I.5 provided Timing routine. This Timing routine automatically keeps track of simulated time and causes the various Events to occur as they are scheduled by the simulation program.

System State Description. The state of a Simulated System is defined in terms of Entities, Attributes of Entities, and Sets of Entities (Ref 29). There are four classes of Entities called Permanent Entities, Temporary Entities, Event Notice Entities, and the Simulated System which can also be considered as an Entity. An Entity is any item or unit that the analyst wishes to independently identify in the system simulation. For example, a job in a processor's input queue. Each type of Entity is described by a list of Attributes. For example, the job in the processor's input queue can have as Attributes, the pointers to the jobs in front and behind it in the queue, the job's identification number, the job's priority, and the pointer to the job's characteristics.

A System's state description may be defined by a number of different types of Entities. Entities are of the same type if their Attribute names are the same; the values cf these Attributes can, of course, be different. A case in point would be the jobs in a host processor's input queue. The analyst must decide which items or units are independently identified in the system simulation as an Entity or an Attribute of another Entity.

Sets of Entities are used to show interrelationships between individual Entities (usually Temporary Entities). For example, all the jobs belonging to a host processor's input queue. These Entities can be inserted or removed from such Sets on a first-in-first-out (FIFO), a last-in-first-out (LIFO), or a ranked basis. When a ranked ordering scheme is used, the Entity's position in the Set is determined by the value of one of its Attributes.

Event and Timing Routines. There are two types of Events used in describing how System state transitions occur. They are; Exogenous Events which are initiated external to the system simulation process by means of an Exogenous Event Tape, and Endogenous Events which are initiated internal to the system simulation process by CAUSE statements in previously executed routines. In a particular system simulation there can be any number of different Events. A particular Event may occur repeatedly and at any point in simulated time. For example, an Event routine can be used to describe how the System's state transitions every time a new job enters a host processor's input queue. Another Event can describe how the System's state transitions when an I/O request interrupt occurs.

Each type of Event (Exogenour and/or Endogenous) appears in an Events List at the beginning of the simulation program. Based upon this list, SIMSCRIPT I.5 automatically generates a Timing routine which keeps track of simulated time and causes the occurrence of the various Events on a next Event scheduled basis. In SIMSCRIPT I.5, simulated time is advanced by variable increments instead of being divided into a series of fixed increments. Thus, Events can occur at any point in simulated time. When the execution of a particular Event is completed, simulated time is advanced to the time of the next scheduled Event, which can be seconds, minutes, hours, or days away, and the Event is automatically caused to occur. The intervening dead time when no System state transitions occur is skipped. The basic unit of time can be set to whatever value the analyst desires (Ref 29).

The various operations performed by the various Event routines or by the subroutines on which they may call can be grouped as follows (Ref 29):

1. Transitioning the state of individual Entities.
2. Causing or cancelling upcoming Events.
3. Executing decision rules.
4. Generating reports.

Transitioning the State of an Entity. Since the state of an Entity is defined in terms of Entities, Attributes, and Sets, an Entity's state can transition in only three ways (Ref 28):

1. The Entity can come into or go out of existence, i.e., be created or destroyed.

2. The Entity can have the value of one or more of its Attributes change.

3. The Entity can change a Set's membership.

SIMSCRIPTI.5 provides the system analyst with a number of language primatives especially constructed for making the above kinds of Entity state transitions. They are. CREATE, DESTROY, LET, FILE, AND REMOVE. A discussion of their use follows.

1. Creating and Destroying an Entity. To create a memory record for a new Entity (a new job) a statement such as CREATE BJOB is used. If the job has completed processing and is leaving the System, a statement such as DESTROY BJOB is used. The DESTROY statement will release the memory record that was reserved by the previous CREATE statement. Only Temporary and Event Notice Entities can be created and destroyed.

2. Changing the Value of an Attribute. The LET statement is used to change an Attribute value. For example, a statement like the following could set the priority of a new job as it enters a host processor's input queue:

   ```
   LET PRIO!(TQUEU) = J
   ```

3. Changing a Set Membership. SIMSCRIPT I.5 provides two language primatives for transferring Entities

into or out of Sets. They are FILE and REMOVE respectively. A statement such as the following would place a new job into the input queue for host processor N.

```
FILE TQUEU IN QUEU(N)
```

A series of statements like the following would check to see if host processor N's input queue is EMPTY, and if not REMOVE the first job in the queue.

```
IF QUEU(N) IS EMPTY, GO TO 80
REMOVE FIRST TQUEU FROM QUEU(N)
```

Causing and Cancelling Events. To maintain control of Event routines, SIMSCRIPT I.5 uses a special type of Temporary Entity called an Event Notice Entity. These Event Notice Entities have the same characteristics as Temporary Entities. That is, they can be created and destroyed, they can have attributes, and they can be members and owners of Sets. The programming steps for causing a future Event routine to occur are (Ref 29):

1. Creating the Event routine's Event Notice.
2. Setting the values of the Event Notice`s Attributes.
3. Scheduling the Event Notice's occurrence.

For example, the arrival of a new job at host processor N, following an inter-arrival time, can be caused by the following series of statements:

```
CREATE BJOB
LET MJOB(BJOB) = N
LET TIAT = -ALOG(RAND * AT(N) * ID
CAUSE BJOB AT TIME + TIAT
```

Prior to its occurrence, the new job arrival can be cancelled by the following statement:

CANCEL BJOB

Executing Decision Rules. Once an Event routine occurs, decisions and/or computations must be made in order to determine how the state of System Entities should change and what future Event routines should be caused or cancelled. For making these decisions and/or computations, SIMSCRIPT I.5 provides the same facilities as does basic FORTRAN. The language allows the calling of both SIMSCRIPT I.5 and FORTRAN subroutines (Ref 28).

Generating Reports. The printed output in SIMSCRIPT I.5 programs is normally generated by a Report Generator routine (Ref 28). The form and content of the output reports are specified on a Report Generator Layout Form by the use of Form Lines and Content Lines. The desired report text is specified in a Form Line. Numerical fields in the Form Line containing one or more numerical fields must be followed by a Content Line which lists the names of the variables which are to be placed in the numerical fields. A number of other features are available from the Report Generator routine which make report generation quite easy (Ref 28).

Data Structure Definition

The term data structure as it is used in this section refers to the manner in which the information describing the state of the general computer network system simulation is maintained or stored. The form of the data structure used in the system simulation is determined by the SIMSCRIPT I.5 programming language. The definition of the data structure, however, is determined by the computer network system model.

The data structure can be divided into five segments according to the type of state information stored or maintained in each segment. The five data structure segments are:

1. Temporary Entity and Attribute information.
2. Permanent Entity and Attribute information.
3. Event Notice Entity and Attribute information.
4. Set and Ranking Attribute information, and
5. Local routine Variable information.

The manner (number of Entities, Attributes, Sets and Local Variables) in which the various segments of information are structured has a direct effect on the size and type of computer network which can be simulated or analyzed by the program. As with the building of the general computer network system model, the implementation of the general computer network system simulation required that a balance be reached between the desires and needs of the analyst and the resources available. A complete listing or glossary of the entire data structure is in Appendix B. A discussion of each of the data structure segments follows.

Temporary Entities and Attributes. The simulation program specifies six Temporary Entities and their Attributes. Five of these Entities correspond to jobs or job segments that are temporarily members of the input, memory, CPU, I/O channel, or output queues discussed in Chapter V. The sixth Temporary Entity is a processor identification number which is used in determining network connectivity.

Permanent Entities and Attributes. There are five types of Permanent Entities specified in the simulation program. They are the host processors, the transmission facilities, the transmission management software, the System workload, and the System itself. These are entities the number of which cannot change during the simulation process. There are some 68 Permanent Attributes divided among the five Permanent Entities. They are used to characterize the computer network system model components and workload described in Chapter V, and to serve as working and performance data gathering areas for the Simulation System.

Event Notice Entities and Attributes. The Event Notice Entities are specifically Endogenous Events only. They do not specify Exogenous Events. The latter type of Event will be discussed later. There are eight Event Notice Entities specified in the general computer network system simulation. The occurrences of the Events specified by these Event Notices are the means by which the System changes state. They also determine how the System will change state. They are the job generators, the job processors, and job transmitters of the simulated network.

Each Event Notice Entity possess at least one Attribute. This Attribute is assigned to each Event by the Timing routine and is the simulated time which the Event is next scheduled to occur.

Sets and Ranking Attributes. Sets depcit interrelationships between individual Entities by grouping and ordering them according to some predetermined scheme. There are six Sets defined in the simulation program. Four of the Sets represent the input, memory, CPU, and I/O channel queues into which jobs or job segments, as Temporary Entities, are placed during their flow through the System. Each of these queues is ranked on highest job priority. The fifth Set serves as the output queue into which jobs are placed upon completing processing at a basis. The sixth Set serves as the first-in-first-out (FIFO) stack used in queues associated with each host processor in the simulated computer network.

Local Routine Variables. Up until this point, all the data structure segments that have been discussed are global to the System. That is, they are System Variables accessible anywhere in the program simply by virtue of their having been declared as Entities, Attributes, or Sets.

Local Variables, on the other hand, are accessible only in the program routine in which they are used. If they are to be accessible in other routines, they must be explicitly passed. Local Variables in SIMSCRIPT I.5 conform to the same rules as do FORTRAN Local Variables.

Simulation Construction Description

The general computer network system simulation is composed of six sections: a Definitions Deck; an Events List; the Events and Subroutines Deck; the Initialization Deck; the Data Deck; and the Exogenous Event Tapes (Ref 29). The specific construction of each section will be explained as each section is discussed. Further clarification of specific details can be found in the User's Manual in Appendix C and the program flowcharts in Appendix A.

Definitions Deck. Each of the Temporary Entities, Permanent Entities, Event Notices, Attributes, and Sets are defined in this section of the simulation program. There is a rigid format which must be observed in constructing or adding to the Definitions Deck (Ref 28).

Events List. The Events List identifies by name, number, and type all of the Exogenous and Endogenous Events contained in the simulation program. This listing of Events is necessary so that the translator can generate an appropriate Timing routine.

Events and Subroutines. This section comprises the bulk of the general computer network system simulation program and is made up of the general computer metwork system simulation program and is made up of the following types of routines:

1. Exogenous Events.
2. Endogenous Events.
3. SIMSCRIPT I.5 subroutines.
4. FORTRAN IV subroutines.
5. SIMSCRIPT I.5 reports.

In the following, each routine is briefly described in terms of its primary functions in the simulation program.

There are three Exogenous Events in the simulation program. These three Events control the execution of a particular simulation run. That is, they initiate the run, specify the System, and terminate the run. The initiation and termination Exogenous Events are necessary in all SIMSCRIPT I.5 simulation programs. The third Exogenous Event is included for convenience and provides a separate facility for initiating the series of Endogenous Events and subroutines involved with job processing at each host processor.

1. Exogenous Event INIT. This Event is the first Event to occur during a simulation run. Its occurrence is triggered by an Exogenous Event Tape and it causes the characteristics of the various computer network and Simulation System components to be read into the simulation program. INIT makes certain tests for correctness on the input data. If the data is found to be incorrectly specified, the simulation run is terminated. Finally, the occurrence of INIT causes the

calculations of network connectivity, reliability, availability, and costs to be performed.

2. Exogenous Event START. As its name implies, this Event causes processing to begin at each of the network host processors on the basis of timing information contained on an Exogenous Event Tape. That is, it causes the *Endogenous Event BJOB to* be scheduled. The START Event also establishes the time at which performance data gathering will begin on the System components which are being observed.

3. Exogenous Event END. This Event is the last Event to occur in a given simulation run. Its occurrence, which is scheduled by an Exogenous Event Tape, signifies the end of the simulation run.

There are eight Endogenous Events in the simulation program. These Events control how the state of the System is to change with the passage of simulated time. That is, they control the flow of jobs through the queueing network described in Chapter V. The Endogenous Events themselves, (their number and functions) were chosen to coincide with the movements of a job or job segment between the various System queues and between the System queues and the host processors' CPUs.

1. Endogenous Event BJOB. This first of eight Endogenous Events generates the jobs which are input to the network through the various host

processors. In doing so, BJOB tests a host processor's input queue to determine if it is full. If the queue is full, the simulation run is terminated. If the input queue is not full, a new job is created, its characteristics are assigned, and it is placed into the proper input queue. After BJOB is initially scheduled by the Event START, it reschedules itself.

2. Endogenous Event RUN. This Event selects the highest priority file request of job segment which is currently awaiting processing at a particular host processor, assign it certain characteristics, and then schedules its execution. The selection process implemented in RUN directly reflects the job selection process described in Chapter V.

3. Endogenous Event TAME. This Event initiates the CPU execution of a particular job segment. It also determines the amount of CPU time that will be used by the job segment prior to its next I/O request.

4. Endogenous Event CPU. This Event assigns the I/O channel to be used by a job segment after the CPU time between I/O request, determined by the Event TAME, has elapsed. If no I/O channels are currently available, it places the job segment into the I/O channel queue.

5. Endogenour Event INTER. This Event occurs after an I/O request has been completed. At that time, INTER compares the priority of the job segment currently in control of a particular host processor's CPU with the priority of the job segment which has just completed the I/O request. Then INTER places the higher priority job segment in control of the host processor's CPU and the remaining job segment it places in the host processor's CPU queue.

6. Endogenous Event NEXT. Upon the completion of a job segment, the Event NEXT determines which job segment will next receive processing by a particular host processor's CPU. The routine first checks the host processor's CPU queue for job segments which had begun processing but were interrupted by I/O requests. If none are found, the routine then checks the host processor's memory queue. If still no job segments are found, the routine selects a new file request from the host processor's input queue to begin processing. A number of different circumstances can occur during this job segment selection process. Such circumstances are handled according to the conventions established for the computer network system model in Chapter V.

If the job segment just completed marks the end of a file request, the routine determines if the file request just completed is the last one associated with a particular job or if the job must be transmitted to another host processor. In the former case, the routine ensures that all System statistics are updated. In the latter case, the NEXT routine assigns a transmission path for the job's use. If not transmission path can be assigned, the job is placed into the particular host processor's output queue.

7. Endogenous Event ENDT. This Event occurs upon the completion of a job transmission. The ENDT routine places the transmitted job into the receiving host processor's input queue. It then determines if there are any jobs waiting transmission which can use the transmission links just made available prior to their being released.

8. Endogenous Event RPT. This Event causes snap-shot performance reports to be made of the System at intervals specified by the analyst.

There are twelve SIMSCRIPT I.5 subroutines in the simulation program. These routines are support routines for the Exogenous and Endogenous Events. The process of determining which Functions would be performed by a subroutine centered around identifying those functions which the analyst might want to change during the course of an analyst project. For example, the transmission management routing algorithm contained in subroutine LINK.

1. SIMSCRIPT I.5 subroutine CHEK. This subroutine determines if the computer network described in the input data is connected. That is, it determines if there is at least one transmission path between any pair of host processors. This routine is an implementation of the connectivity algorithm described in Chapter V. It uses Entities, Attributes, and Sets as described earlier in this chapter.
2. SIMSCRIPT I.5 subroutine CALCOST. This subroutine calculates the total monthly cost of the computer network configuration being simulated.
3. SIMSCRIPT I.5 subroutine AREA. This subroutine determines the portion of the data structure in which the characteristics of a newly created job will be placed. If the amount of storage area set aside in the data structure for job characteristics is full, the simulation run is terminated.
4. SIMSCRIPT I.5 subroutine ASIGN. This subroutine assigns a job type to a newly created job. It also adds a list of all the host processors associated with the job type to the new job's characteristics.
5. SIMSCRIPT I.5 subroutine IO. This subroutine describes an I/O request that has been made by a particular job segment and then schedules the

actual I/O interrupt itself.

6. SIMSCRIPT I.5 subroutine ENDSG. Upon the completion of a particular job file request, ENDGS updates the various statistics being gathered in association with job file requests.

7. SIMSCRIPT I.5 subroutine ENDJB. This subroutine updates the various statistics associated with job (job type) completions.

8. SIMSCRIPT I.5 subroutine RELSA. This subroutine makes that portion of the data structure which was being used by the just completed job available for use by a newly created job.

9. SIMSCRIPT I.5 subroutine LINK. This subroutine implements the fixed-minimum-path routing algorithm discussed in Chapter V. That is, it searches for the shortest transmission path between two host processors. If the number of links-traversed by the minimum path found exceeds five, the routine states that a transmission path is not currently available.

10. SIMSCRIPT I.5 subroutine ACCT. This subroutine updates the transmission processing statistics for the transmission request just initialed. It also schedules the end of the transmission.

11. SIMSCRIPT I.5 subroutine RESL. This subroutine releases the transmission links just used and computes the total usage time for each of the

transmission links.

12. SIMSCRIPT I.5 subroutine STATS. This subroutine first complies the performance data pertinent to the performance measures being made on the simulated computer network. It then places these reduced data items into a subscripted Permanent Attribute for output.

There are three FORTRAN IV subroutines in the simulation program. All three subroutines are associated with the production of the Network Reliability/Availability Report. The subroutines are written in FORTRAN IV, but could just as easily be written in SIMSCRIPT I.5. As with the SIMSCRIPT I.5 subroutines, the functions selected for each subroutine allow the analyst to make functional changes with a minimum amount of effort.

1. FORTRAN IV subroutine NETRA. This subroutine calculates the reliability of successful communication and the availability of communication for each pair of host processors in the computer network being simulated. It also calculates a pessimistic and mean reliability and availability percentage for the computer network as a whole. This routine reads its own data and outputs its own reliability/availability reports.
2. FORTRAN IV subroutine PATH. This subroutine determines the transmission paths or tie sets between a particular pair of host processors.

It is used in conjunction with the NETRA subroutime.

3. FORTRAN IV subroutine CALAVAL. This subroutine calculates the percent availability for each individual transmission path between two host processors. It is used in conjunction with the NETRA subroutine.

There are eleven SIMSCRIPT I.5 Report routines in the simulation program. With the exception of the reliability/availability reports, all program output is handled through these routines. The format of those reports is illustrated in Appendix D.

1. SIMSCRIPT I.5 Report RPORT 1. RPORT 1 reports the input data associated with each host processor. There is a separate report generated for each host processor.
2. SIMSCRIPT I.5 Report RPORT 2. RPORT 2 reports the input data associated with each transmission link. It also reports a calculated monthly cost for each link. There is one consolidated report for all the transmission links.
3. SIMSCRIPT I.5 Report RPORT 3. RPORT 3 reports the calculated basic monthly configuration costs for the computer network configuration being simulated. There is one such report produced per simulation run.
4. SIMSCRIPT I.5 Report REPORT 1. REPORT 1 is a

periodic performance report on each host processor in the network. There is a separate report produced for each host processor. The interval between reports is specified by the analyst.

5. SIMSCRIPT I.5 Report REPORT 2. REPORT 2 is a periodic performance report on each transmission link in the network. There is one consolidated report for all the transmission links. The interval between reports if specified by the analyst and coincides with the interval between host processor performance reports.

6. SIMSCRIPT I.5 Report REPORT 3. REPORT 3 is a periodic performance report on each of the I/O channels for each host processor in the network. There is a separate report produced for each host processor. The report interval is the same as that for the REPORT 1 and REPORT 2 reports.

7. SIMSCRIPT I.5 Report BANNER. Upon successful termination of a particular simulation run, BANNER prints that the simulation was successfully terminated and the time at which termination occurred.

The remaining four Report routines are used for the reporting of errors and for the detailed tracking of program occurrences. The errors identified by the simulation are ones which result from either incorrectly specified input data or from situations which can occur during a simulation run and which would result in useless results. The errors reported are, by number, the following:

1. The interval between performance reports is equal to zero.
2. The computer network topology specified is not connected.
3. The job priority input data for host processor N is incorrectly specified.
4. The number of job segments input data for host processor N is incorrectly specified.
5. The maximum number of jobs allowable in the network has been exceeded (i.e., the job characteristics storage area of the data structure if full).
6. The job type input data is incorrectly specified.
7. The I/O channel assignment routines are in error.
8. The transmission link assignment routines are in error.
9. The maximum allowable input queue length for host processor N has been exceeded.
10. The job type cumulative probability data is in error.

The occurrence of any of the above errors causes the simulation run to terminate.

8. SIMSCRIPT I.5 Report ERROR 1. For those errors in which a reference to a specific host processor would have no meaning, ERROR 1 prints the error number, the simulated time when the error occurred, and the number of the routine in which it occurred.

9. SIMSCRIPT I.5 Report ERROR 2. For those errors in which a reference to a specific host processor would aid in diagnosing the error, ERROR 2 prints

the error number, the simulated time when the error occurred, the number of the routine in which it occurred, and the number of the associated host processor.

10. SIMSCRIPT I.5 Report PARAMS. This report provides a means of tracking the occurrence of every Event and subroutine during a given simulation run. The report itself contains the number of the routine, the simulated time when it occurred, and certain pertinent data items. The report is triggered on and off by the analyst.

11. SIMSCRIPT I.5 Report PRIODS. This report is used in conjunction with the report PARAMS. It provides detailed information on the characteristics of the jobs being processed by the network host processors.

Initialization Deck. The Initialization Deck provides initial values to Permanent Attributes, defines how many Permanent Entities are associated with each Permanent Attribute, and assigns Permanent Attributes to Permanent Entities. The deck itself consists of one System Specification Card followed by initialization cards and lastly one blank card (Ref 28).

The use of the Initialization Deck allows the analyst to either initialize the System to an idle or empty state as is done in the general computer network system simulation, or to initialize the System to some state in its actual life cycle.

Data Deck. The Data Deck contains all the information necessary to characterize the computer network to be simulated and its environment. The structure of the Data Deck is shown in Appendix C. The deck is composed of four groups of data: host processor specification data; topology specification data; reliability/availability data; and transmission link specification data. The host processor data includes the workload characterization for each host processor in the network.

Summary

The fifth step in the development of the general computer network system simulation was to translate the computer network system model described in Chapter V into the simulation programming language SIMSCRIPT I.5. In doing so, the computer network system model was described in terms of Permanent and Temporary Entities, Permanent and Temporary Attributes, Sets of Entities, Exogenous and Endogenous Events, and SIMSCRIPT I.5 and FORTRAN IV subroutines. Such a description also determined both the data structure that would be used to support the simulation and the simulation construction.

VII. SIMULATION EXPERIMENTS

Introduction

The final phase in the development of the general computer network system simulation was to conduct a series of simulation experiments which would establish to what extent the system simulation reflected the general computer network system model described in Chapter V. This experimentation phase began when the basic Exogenous and Endogenous Event routines were first integrated into a workable simulation program. The simulation experiments, of course, served to indicate those aspects of the evolving simulation program which were in error as well as those which were correct. Thus, simulation experimentation as it was applied to the development of the general computer network system simulation was an iterative process which sought to eliminate logic errors and to tune the system simulation to the system model.

The simulation experiments which were conducted on the general computer network system simulation can be divided into three categories. The categories are:

1. Job Flow experiments,
2. Performance measure and error exercising experiments, and
3. Computer network experiments.

In this chapter, each of the three categories of experimentation is discussed along with the pertinent results of each.

Job Flow Experiments

This category of simulation experiments sought to trace in detail the job flow in the individual events, subroutines, and reports as well as in the overall system simulation. This was done in order to determine if the job flow represented in the simulation was in fact the job flow described by the general computer network system model.

The foundation for this category of experimentation was the fact that each job in the simulated network possessed a unique job number. This job number was assigned to a new job when it entered the network and was retained until the job completed its processing. Making use of SIMSCRIPT I.5 Reports PARAMS and PRIODS, discussed in Chapter VI, it was possible to trace a new job from its creation until it was destroyed. These trace reports provided such information as the job's number, the routines the job exercised, and the simulated times at which the job exercised the routines.

The job flow experiments were controlled by limiting the number of job segments per job file request and by structuring the workload characterization of the host processors. For example, the number of job segments per job file request was limited to one. Additionally, all but one of the host processors' workloads were limited to a single job type requiring processing only from the originating host processor. The remaining host processor's workload was limited to a single job type requiring processing at all of the host processors in the simulated network. Controlling the experiments in this manner greatly facilitated the use of the job trace information obtained during simulation runs in tracing individual jobs.

A large number of simulation experiments were conducted in this category. Analysis of the job trace data obtained indicated that the job flow described by the general computer network system simulation was a logically accurate reflection of the job flow described by the general computer network system model.

Performance Measure and Error

As its name implies, this category of simulation experiments was directed toward exercising all of the various performance measures and errors of the simulation. There were two groups of features examined during these experiments. The first group were the various error messages that had been incorporated into the simulation. The method of exercising this group of features was quite simple. An error was introduced and its detection and reporting by the simulation was noted. For example, the new job interarrival times at the host processors were shortened disproportionately to the service times of the various system resources (i.e., the CPU, transmission facilities, etc.). This caused the maximum number of jobs allowable in the network at one time to be exceeded and an error message to that effect to be output. Each of the possible error situations was created in a similar manner and the simulation response (error message and simulation termination) was observed.

The second group of features to be exercised were the general computer network system simulation's performance measures. As with the first group of features, situations were created which would cause a particular performance measure to react in a predetermined manner. The predicted behavior was then compared with the observed behavior following the simulation run. For example, a situation similar to the one described in the discussion on the job flow experiments was created in order to observe the host processor input queue measures. The deviation from the described situation was that one of the host processors was not allowed to start processing jobs. This situation should cause the nonoperative host processor's input queue to become filled with only those jobs transmitted to it from elsewhere in the network. It should also cause the maximum input queue length and the current input queue length to be equal. A simulation run was conducted and the measured behavior was seen to approximate the predicted behavior of the performance measures. Each of the performance measures made on the simulated system were exercised in a similar way and seen to behave as predicted.

System Experiments

The third and final category of simulation experiments was concerned with demonstrating the performance of the general computer network system simulation in an evaluation of alternatives role as discussed in Chapter II. Two sources of variation upon a basic system characterization were used. The first involved the workload (job types) presented to each of the host processors and the second concerned the type of global topology (transmission links) interconnecting the host processors. The basic system characterization was composed of four host processors interconnected in a ring type global topology (See Figure 9). The characteristics used to describe the host processors, the workload, the transmission links, the reliability and availability of a transmission link, and the configuration costs were selected based upon two criteria. The first criteria was that the characteristic be realistic. The second criteria was that the characteristics (such as job inter-arrival times) allow the simulated system to reach a steady state (i.e., approximately as many jobs would be completed as would be input over a given interval of time) in a relatively short period of simulated time.

Five basic system experiments were conducted. They each ran for a simulated six hours, with interval performance reports, being made every two hours. This timing of the simulation and of the reports was chosen after several preliminary runs indicated that the simulated system

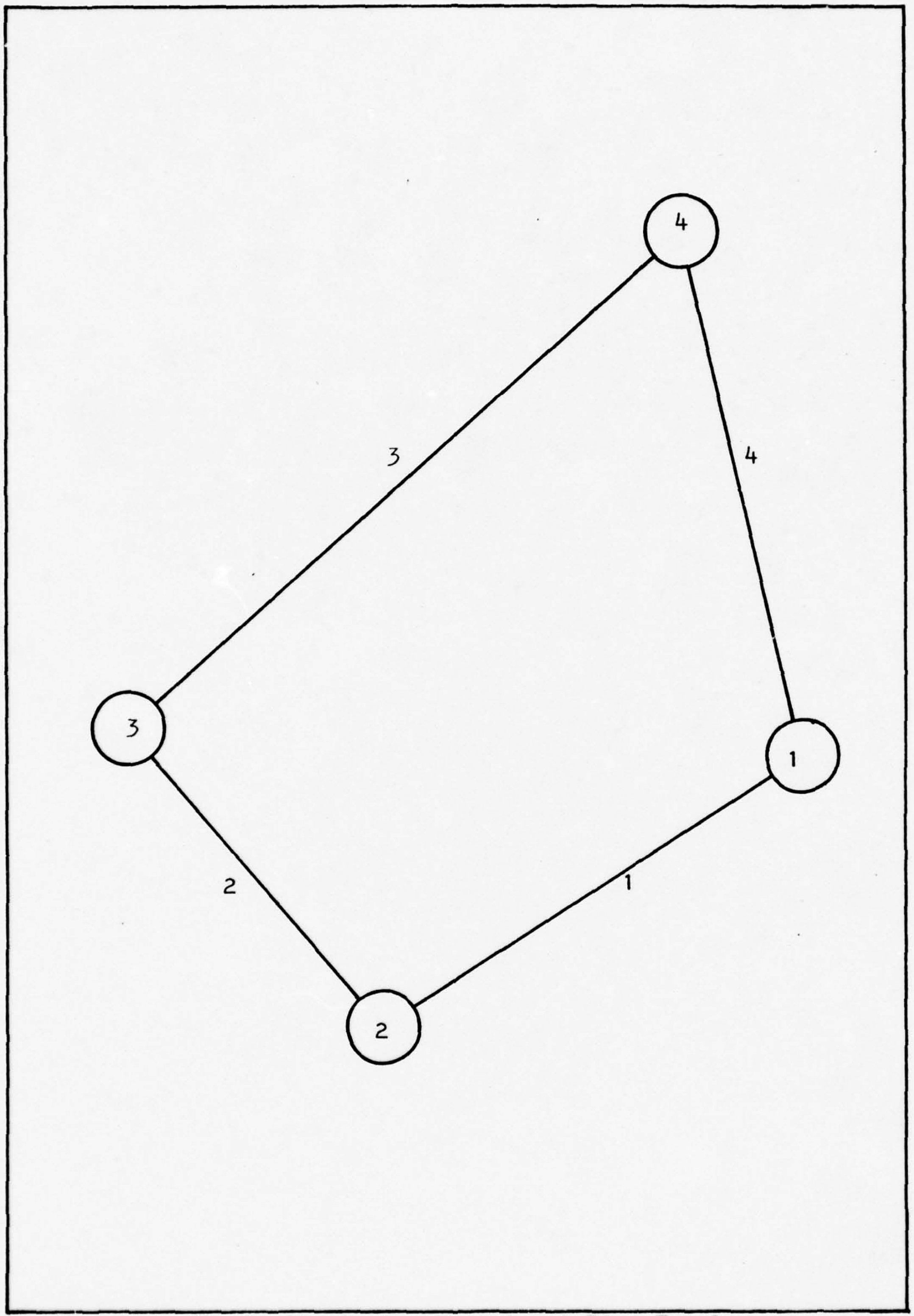


Fig 9. Simulation Experiment Ring Topology

reached a steady state before two hours of simulated time had elapsed. The basic unit of time in each experiment was a second. The use of a second as the basic unit of time was motivated by the fact that most computer system accounting data uses the second as the basic unit of time. The first two experiments were concerned with workload variations. The last three experiments studied the effects of variations in the global computer network topology. The characterization and performance data for each experiment is contained in Appendix D.

Experiment I. In this experiment, none of the jobs input at the host processors required processing at any host processor other than the one at which the jobs were input. Thus, the number of job file requests associated with a job was limited to one. Additionally, the number of job segments per job file request was limited to one in all cases except that of host processor three, where there was an equal probability of a file request having from one to ten job segments. Finally, the priority assigned to a new job was restricted to priority one for all host processors except host processor two. In the case of processor two, a new job had an equal probability of having a priority from one to ten.

A number of major results could be predicted from the above workload characterization. First, the number of file requests processed at a host processor should equal the number of jobs completed at the same host processor.

Additionally, the mean throughput time per file request should equal the mean throughput time per job. Second, there should be no transmission requests originated and consequently no transmission link utilization statistics produces. Third, the processor utilization of host processor three should be higher than that of the other host processors due to the large number of job segments requiring CPU execution time. The larger number of job segments at host processor three should also cause an increased number of I/O requests to be processed and thus show a higher I/O channel utilization. Finally, the job priority statistics for host processor two should indicate the processing of file requests having priorities from one to ten. The results of Experiment I, as shown in Appendix D, confirmed all the predicted results discussed above.

Experiment II. This experiment was very similar to the previous experiment in the manner in which the jobs input at the host processors were characterized. The dissimilarities, however, were that all of the jobs input at the host processors would have priorities of one, that all of the job file requests processed would have only one job segment, and that all of the new jobs input at host processor one would require processing at each of the host processors in the simulated network.

As was the case with the prior experiment, a number of major results could be predicted from the manner in which the network workload had been characterized. First, the number of job file requests processed by a host processor would be greater than the number of jobs completed by the same host processor. In fact, the difference between the two should be approximately equal to the number of jobs completed by host processor one. This would be due to the fact that all of the host processors except host processor one would be required to process a file request belonging to a job originated at host processor one and that host processor one would be able to show job completion only after a job had received processing from each of the other host processors. Additionally, the mean throughput time per job should be greater than the mean throughput time per file request. Second, the number of transmission requests originated at each host processor should equal or be slightly greater than the number of jobs completed by host processor one. This again would be due to distributed processing required by jobs input at host processor one. Finally, the individual host processors should show a general increase in processor utilization over Experiment I.

The results of Experiment II, as shown in Appendix D, confirmed all of the predictions concerning the performance of the simulated network. One interesting result of the experiment was noted. It was that the number of tuple requests between a given pair of host processors did not equal the number of transmission requests originated between the same pair of host processors. This result was due to the fixed-minimum-path routing algorithm selecting a transmission path other than the most direct for the transmission of a job.

Experiment III. This experiment was the first in a series of three experiments where the source of variation between experiments would be the global network topology. All other aspects of computer network characterization would remain constant.

Experiment III was configured in a ring topology (See Figure 9) and was to serve as the basis for comparison with the results of Experiments IV and V. In Experiment III as well as Experiments IV and V, the jobs input at the host processors had an equal probability of requiring processing at one, two, three, or four of the host processors. In addition, the number of job segments per job file request was limited to one. Various probabilities for job priorities were assigned to each host processor.

This series of experiments also contained calculations of network reliability and availability. These calculations would provide yet another means of comparing the different topologies characterized by the individual experiments.

The results of Experiment III were expected to generally follow those of Experiment II, except that a general increase in the numbers was anticipated. This was because of the increased number of jobs requiring transmission through the network. The results of the actual experiment seemed to confirm the anticipated results.

Experiment IV. This experiment was the second in the series of three experiments where the source of variation was the global network topology. In this particular experiment, the ring topology of Experiment III had been changed to the centralized global topology shown in Figure 10.

Certain predictions concerning the results of this system experiment were made. First, due to the decreased number of transmission links as well as their configuration, the number of transmission requests handled by any given transmission link should increase. Additionally, the maximum output queue length, the mean output queue length, and the mean transmission delay time should increase. Second, the processor utilization, the number of file request completed, and the number of jobs completed should decrease. This was because it was felt that the additional transmission delay would decrease the amount of work completed in the six hour observation period. Finally, the reliability and availability of communication throughput the computer network should in general decrease.

The actual results of the simulation run confirmed all but one of the predicted results. While the mean transmission delay time did increase, it did not seem to have a substantial effect on the number of jobs completed. The delay was apparently not long enough to have the anticipated effect.

Experiment V. Experiment V was the third and last in the series of experiments where the source of variation

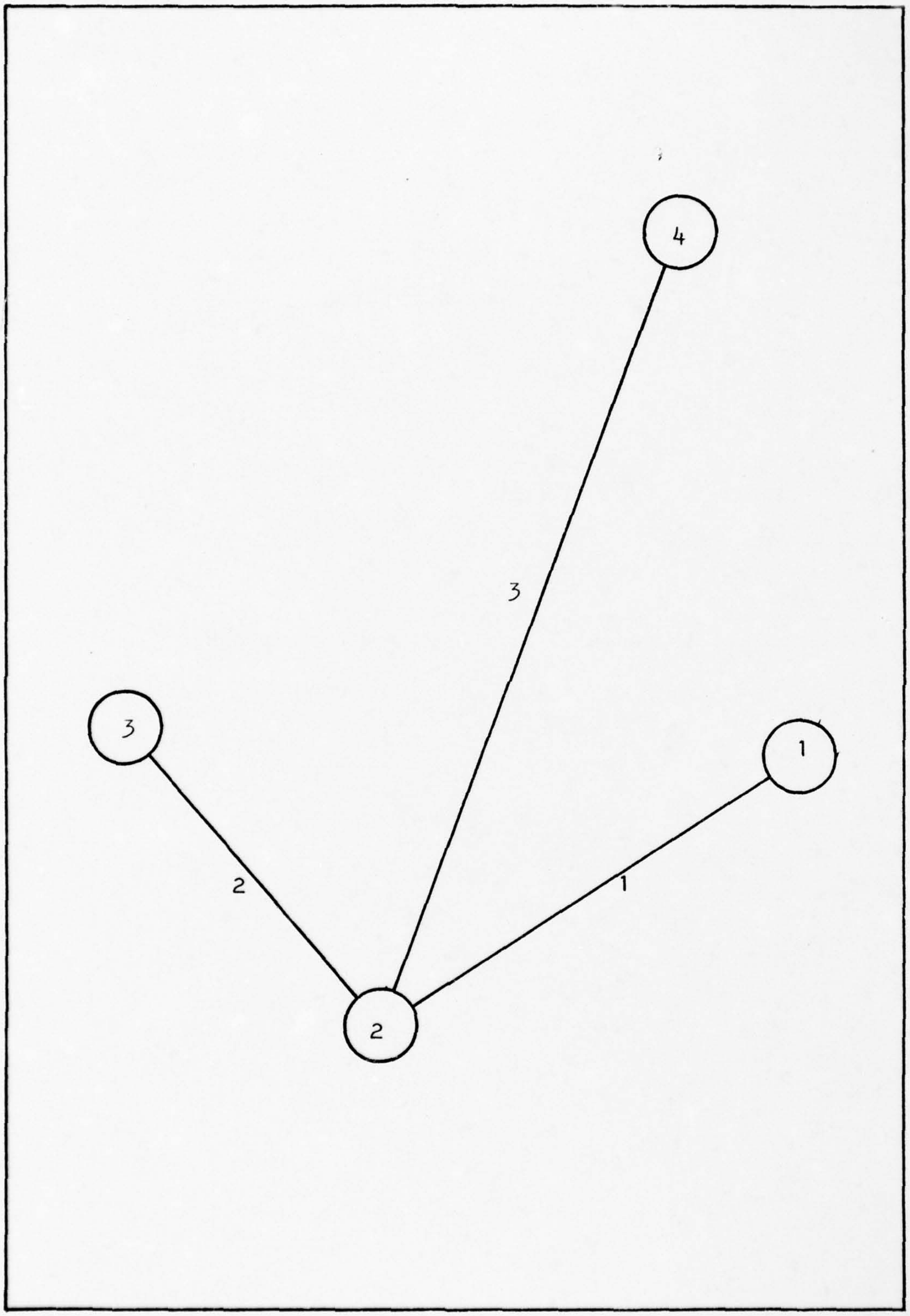


Fig 10. Simulation Experiment Centralized Topology

was the global network topology. In Experiment V the ring and centralized topologies of Experiments III and IV were combined into the distributed global topology shown in Figure 11.

As in all the previous experiments, certain predictions were made about the performance of the simulated system. First, as a result of the increased number of transmission links as well as the distributed topology the number of transmission requests sent over any given transmission link should decrease. In addition, the maximum output queue length, the mean output queue length, and the mean transmission delay time should all decrease. Second, the processor utilization, the number of file request completed, and the number of jobs completed should all increase. This was because with the decreased mean transmission delay time, the number of jobs able to flow through the computer network should increase. Correspondingly, the reliability and availability of communication throughout the network should in general increase as a result of the increased number of possible transmission paths between host processors.

The results of this final system experiment agreed with the anticipated results. The overall results of the last three simulation experiments indicated that the general computer network system simulation could be used to obtain a "feel" for different topologies applied to a given computer network environment.

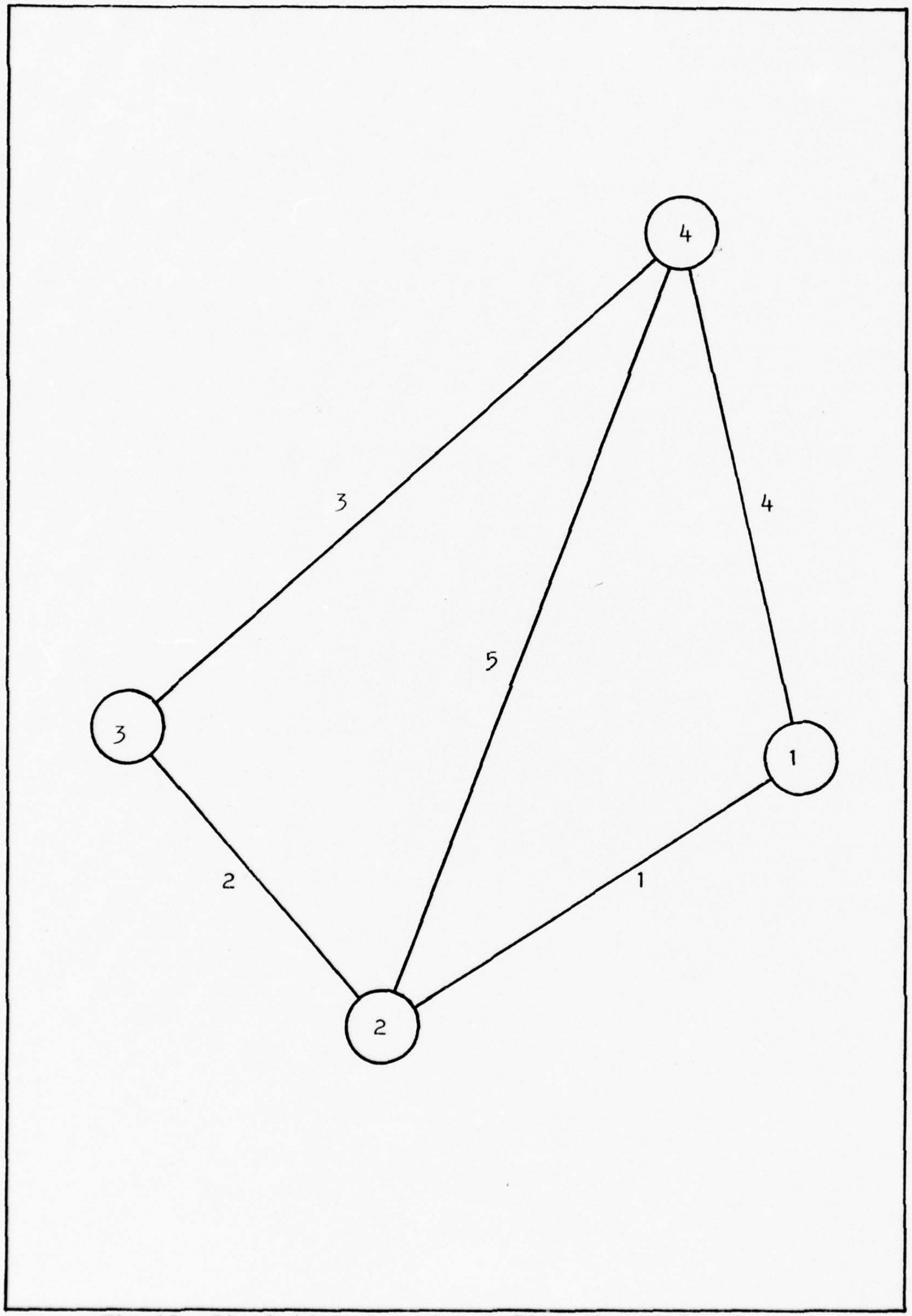


Fig 11. Simulation Experiment Distributed Topology

Summary

Three categories of simulation experiments were conducted on the general computer network system simulation. They were:

1. Job flow experiments,
2. Performance Measure and Error exercising experiments, and
3. Computer Network experiments.

These experiments sought to eliminate errors in the system simulation and to tune it to the general computer network system model. Thus, the simulation experimentation process itself was an iterative process which resulted in a general computer network system simulation which it is felt accurately reflects the computer network system model described in Chapter V. However, it must be emphasized that the performance of the system model has not been compared with the performance of an actual computer network and that caution must be exercised in using the system simulation.

VIII. CONCLUSIONS AND RECOMMENDATIONS

Introduction

This chapter discussed several conclusions resulting from the development of a general computer network simulation model. Recommendations for further investigations and expansions of this simulation model are also discussed.

Conclusion

A general computer network system model has been developed. This model has been implemented in the discrete event simulation programming language SIMSCRIPT I.5. It is felt that within the constraints mentioned in Chapters V and VI the general computer network simulation model is capable of effectively aiding in the evaluation of the following aspects of a computer network's performance.

1. Reliability.
2. Availability.
3. Survivability.
4. Network Effectiveness and Efficiency.
5. Basic monthly costs.
6. Data-base and applications software distribution.

It should be emphasized that the general computer network simulation model has not been subjected to a formal verification and validation. That is, the performance of the simulation model has not been compared with the performance of an existing computer network. However, in its present state of development it is felt the simulation model is adequate for obtaining a basic "feel" or understanding of computer network performance.

The development of the general computer network simulation model, like the development of any other product, went through a series of clearly definable but overlapping and frequently iterative steps. It is felt that these steps as they evolved are of a general nature and can be applied to the development of most complex system simulations models. The steps in this development process are as follows:

1. Study system simulation in general.
2. Study the system to be modeled and simulated in general.
3. Study the performance measures that can be made on the system in general.
4. Design a system model.
5. Implement the system model.
6. Conduct simulation experiments.

The primary difference in this approach versus others that have been proposed is that it explicitly includes a study of the performance measures that can and should be made on the system to be studied. It is felt that to include such a study prior to the design of the system model should aid in the elimination of unnecessary modeling detail.

Recommendations

There are three areas in which the general computer network simulation model could be expanded and enhanced. The first would involve a more detailed characterization of the I/O requests and the I/O devices at each of the host processors. This addition to the simulation model would require that a method of characterizing I/O requests similar to that used to characterize jobs be developed. It would also require the addition of routines to simulate the behavior of each type of I/O device associated with a particular host processor. This change would allow the analyst to characterize the host processors more realistically.

The second enhancement that could be made to the simulation model calls for the inclusion of a packet-switching capability for message transmission. This change would require that the SIMSCRIPT I.5 subroutine LINK be expanded to include the selection of an adaptive routing algorithm. It would also require that the SIMSCRIPT I.5 subroutine ACCT be expanded to handle the division of messages into packets before transmission. This change would allow the analyst to have the option of simulating either a packet-switching or a message-switched computer network.

The third enhancement would expand the manner in which the simulation model could be used. It calls for the formal verification and validation of the simulation model with data from an existing computer network. This enhancement would require that a sizeable workload characterization and performance measurement and evaluation study be performed on the existing network. Such an effort would greatly enhance the value of the general computer network simulation model.

BIBLIOGRAPHY

BIBLIOGRAPHY

1. Frank, H. and W. Chou. "Topological Optimization of Computer Networks." Proceedings of IEEE, 61:1385-1397 (November 1972).

2. Air Force Data Systems Design Center. STALOG Study Report. Gunter Air Force Station, Alabama, 15 June 1973.

3. Flagle, C.D., et al. Operations Research and Systems Engineering. Baltimore: The John Hopkins Press (1960).

4. Fishman, G.S. Concepts and Methods in Discrete Event Simulation. New York: John Wiley & Sons (1973).

5. Kiviat, P.J. Computer Simulation: Modeling Concepts. Santa Monica, California: The Rand Corporation, RM-5378-PR (1967).

6. Teichroew, D., et al. "Computer Simulation: Discussion of the Technique and Comparison of Languages." Communications of the ACM, 9 No. 10: 723-741 (1966).

7. Pierce, J.R. "How Far Can Data Loops Go." IEEE Transactions on Communications, COM-20 No. 3: 527-530 (1972).

8. Davies, D.W. and D.L.A. Barber. Communication Networks For Computers. London: John Wiley & Sons (1973).

9. Roberts, L.G. and B. Wessler. "The ARPA Computer Network Development to Achieve Resource Sharing." Proceedings of AFIPS Spring Joint Computer Conference, 38: 1408-1423 (1972).

10. IBM. "IBM 3600 Finance Communication System-System Summary." GC27-0001-2. (1973).

11. Hayes, J.F. and D.N. Sherman. Bell System Technical Journal, Vol 50, No. 9:2947-2978 (November 1971).

12. Pack, C.D. "The Effects of Multiplexing on a Computer-Communications System." Communication of the ACM, Vol. 16, No. 3: 161-168 (March 1973).

13. Doll, D.R. "Multiplexing and Concentration." Proceedings of IEEE, Vol. 60, No. 11: 1313-1321 (November 1972).

14. Chou, W. "Computer Communications Networks-The Parts Make Up the Whole." *Proceedings of AFIPS National Computer Conference*, *Vol. 44* 119-128 (1975).

15. Crocker, S.D., *et al*. "Function-Oriented Protocols for the ARPA Computer Network." *Proceedings of AFIPS Spring Joint Computer Conference*, *Vol. 38* 271-279 (1972).

16. Kleinrock, L. *Queueing Systems*. Vol. II: *Computer Applications*. New York: Wiley Interscience (1976).

17. Grubb, D.S. and I.W. Cotton. *Criteria For The Performance Evaluation of Data Communications for Computer Networks*, NBSTN-882. Washington: Government Printing Office. (1975).

18. Yourdon, Edward. *Design of On-Line Computer Systems*. Prentice-Hall, Inc. Englewood Cliffs, New Jersey (1972).

19. Svobodova, L. *Computer Performance Measurement and Evaluation Methods: Analysis and Applications*. New York: American Elsevier Publishing Company, Inc. (1976).

20. Gerla, M. "New Line Tariffs and Their Impact on Network Design." *Proceedings of AFIPS National Computer Conference*, *Vol. 43*, 577-582 (1974).

21. Loomis, D.C. "Ring Communication Protocols." Unpublished Technical Note #26, 1973 pp 1-54.

22. Redding, J.L. *Computer Network Simulator*. Naval Ship Research and Development Center (Ad-730 053), 1971.

23. Lehman, M. and J. Rosenfeld. "Performance of a Simulated Multiprogramming System." *Proceedings of AFIPS Fall Joint Computer Conference*, *Vol. 33*, *Part II*, 143-1442 (1968).

24. Anderson, H.A. and R.C. Sargent. "A Statistical Evaluation of the Scheduler of an Experimental Interactive Computer System." *Statistical Computer Performance Evaluation* (Editor: W. Freidberger): Academic Press, 73-98 (1972).

25. Nelson, A.C. Jr., *et al*. "A Computer Program for Approximating System Reliability." *IEEE Transactions on Reliability*. *R-19*, *No. 2*, 61-65 (1970).

26. Messinger, M. and M.L. Shooman. "Reliability Approximations for Complex Structures." Proceedings IEEE Annual Symposium on Reliability, 292-304 (1967).

27. Regulinski, T.L. Optimization of Reliability Processes. Ph.D. Dissertation. Bradford, England: University of Bradford, 1976.

28. Control Data Corporation, SIMSCRIPT Reference Manual 6000 Version 3, Publication Nimber 60358500. Minnespolis, Minnesota: Control Data Corporation.

29. Markowitz, H.M. et al. SIMSCRIPT A Simulation Programming Language. Englewood Cliffs, N.J.: Prentice-Hall (1963).

SUPPLEMENTAL BIBLIOGRAPHY

SUPPLEMENTAL BIBLIOGRAPHY

1. Abrams, M. Computer Networks: Text and References For a Tutorial. New York: IEEE Press (1976).

2. Abramson, N. and F.F. Kuo. Computer-Communications Networks. Englewood Cliff, New Jersey: Prentice-Hall (1973).

3. Badenius, D. "Failure Rate/MTBF." IEEE Transactions on Reliability, R-19, No. 2, GG-67 (1970).

4. Bell, T.E. "Objectives and Problems in Simulating Computers." Proceedings of AFIPS Fall Joint Computer Conference, Vol. 39, 287-297 (1972).

5. Bellman, J.A. "Railroad Network Model." Simulation, Vol. 11 5-13 (1969).

6. Blanc, R.P. and I.W. Cotton. Computer Networking. New York: IEEE Press (1976).

7. Bowdon, E.K. St., et al. "Simulation-A Tool for Performance Evaluation in Network Computers." Proceedings of AFIPS National Computer Conference, Vol. 42, 121-131 (1973).

8. Chou, W. et al. Simulation of Centralized Computer Communications Systems. Glen Cove, New York: Network Analysis Corporation (1976).

9. Cole, G.D. Computer Network Measurements: Techniques and Experiments. Ph.D. Dissertation. University of California, Los Angeles (1970).

10. Conway, R.W. "Some Tactical Problems in Digital Simulation." Management Science, Vol. 10, No. 1, 47-61 (1963).

11. Dotson, W.P. An Analysis and Optimization for Probabilistic Graphs. Ph.D. Dissertation. Air Force Institute of Technology, Ohio (1976).

12. Frank, H. "Computer Networks: Art to Science to Art." Networks, Vol. 5, 7-32 (1975).

13. Frank, H. "Computer Netwoeks." Networks, Vol 5, 69-73 (1975).

14. Frank, H., et al. "Optimal Design of Centralized Computer Networks." Networks, Vol. 1, 43-57 (1971).

15. Frank, H., et al. "Topological Considerations in the Design of the ARPA Computer Networks." Proceedings of AFIPS Spring Joint Computer Conference, Vol. 36, 581-587 (1970).

16. Frank, H. and W. Chou. "Network Proper Ties of the ARPA Computer Network." Networks, Vol. 4, 213-239 (1974).

17. Frank, H. and I.T. Frisch. "Analysis and Design of Survivable Networks." IEEE Transactions on Communication Technology, Com-18, No. 5, 501-519 (1970).

18. Frank, H. and I.T. Frisch. Communication, Transmission, and Transportation Networks. Reading, Massachusetts: Addison-Wesley (1971).

19. Gagne, R.E. "Computer Modeling and Simulation Handbook...MK III." Simulation, Vol. 26, 147-154 (1976).

20. Gerla, M. The Design of Store-And-Forward (S/F) Networks For Computer Communications. Ph.D. Dissertation, University of California, Los Angeles (1973).

21. Hansler, E., et al. "Exact Calculation of Computer Network Reliability." Networks, Vol. 4, 95-112 (1974).

22. Hiller, F.S. and G.J. Lieberman. Introduction to Operations Research. San Francisco: Holden-Day. (1967).

23. IEEE Computer Society. "Trends and Applications 1976: Computer Networks." 76 CHII43-7C. New York, (1976).

24. Jensen, P.A. and M. Bellmore. "An Algorithm to Determine the Reliability of a Complex System." IEEE Transactions on Reliability, R-18, 169-174 (1969).

25. Kleinrock, L. "Analytic and Simulation Methods in Computer Network Design." Proceedings of AFIPS Spring Joint Computer Conference, Vol. 36, 569-579 (1970).

26. McGregor, P. et al. Communications Processor Simulation: A Practical Approach. Glen Cove, New York: Network Analysis Corp. (1976).

27. Pooch, V.W., and H.E. Livings. "A Heuristic Teleprocessing Network Design." Proceedings of AFIPS Spring Joint Computer Conference, Vol. 36, 569-579 (1970).

28. Price, W.L. Graphs and Networks An Introduction. London: Butterworth and Co. Ltd. (1971).

29. Rubin, I. "multiterminal Path Flows for Packet-Switching Communication Networks." Networks, Vol. 5. 205-235 (1975).

30. Rustin, R. Computer Networks. Englewood Cliffs, New Jersey. Prentice Hall (1972).

31. Shooman M.L. Probabilistic Reliability: An Engineering Approach. New York: McGraw-Hill (1968).

32. Urban, G.L. "Building Models for Decision Makers." Interfaces, Vol. 4, No. 5, 1-11 (1974).

33. Van Slyke, R., et al. "Avoiding Simulation in Simulating Computer Communication Networks." Proceedings of AFIPS National Computer Conference, Vol. 42, 165-169 (1973).

34. Wilkov, R.S. "Analysis and Design of Reliable Computer Networks." *IEEE Transactions on Communications*, COM-20, 660-678 (1972).

35. Wilkov, R.S., et al. "Topological Considerations in Computer Network Design." Proceedings Fourth Data Communications Symposium, 4-1-4-24 (1975).

36. Wyman, F.P. Simulation Modeling: A Guide to Using SIMSCRIPT. New York: John Wiley & Sons, (1970).

APPENDIX A

GENERAL COMPUTER NETWORK SYSTEM SIMULATION FLOWCHARTS

APPENDIX A

General Computer Network System Simulation Flowcharts

Introduction

A symbolic flowchart provides a convenient way of describing the functions of the various routines of the system simulation. The symbolic conventions used in this appendix are shown in Figure A-1. They are based upon the convention set forth by P.J. Kiviat in Digital Computer Simulation: Modeling Concepts, The Rand Corporation RM-5378-PR, Santa Monica, California, 1967. The system simulation flowcharts follow these two pages.

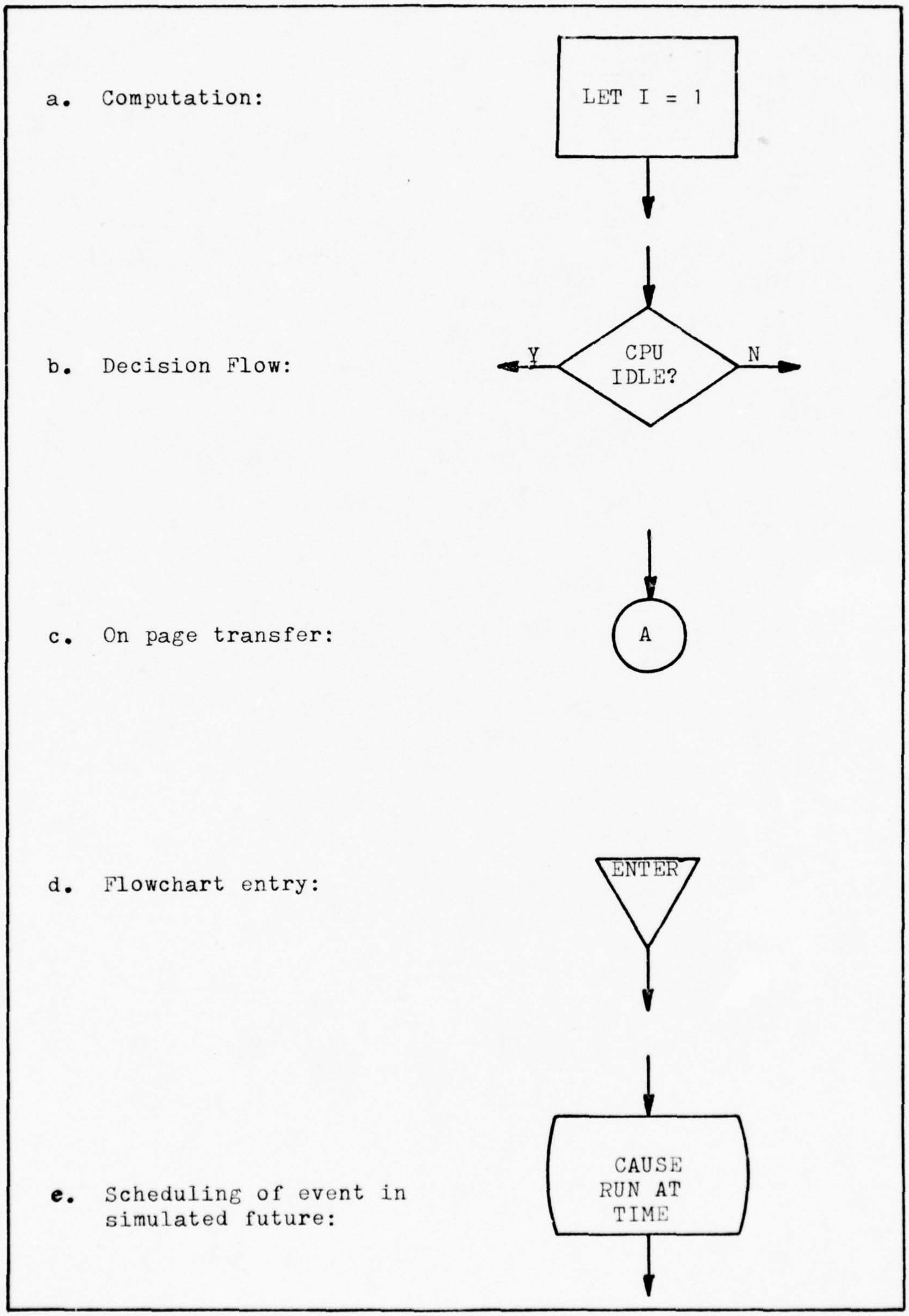
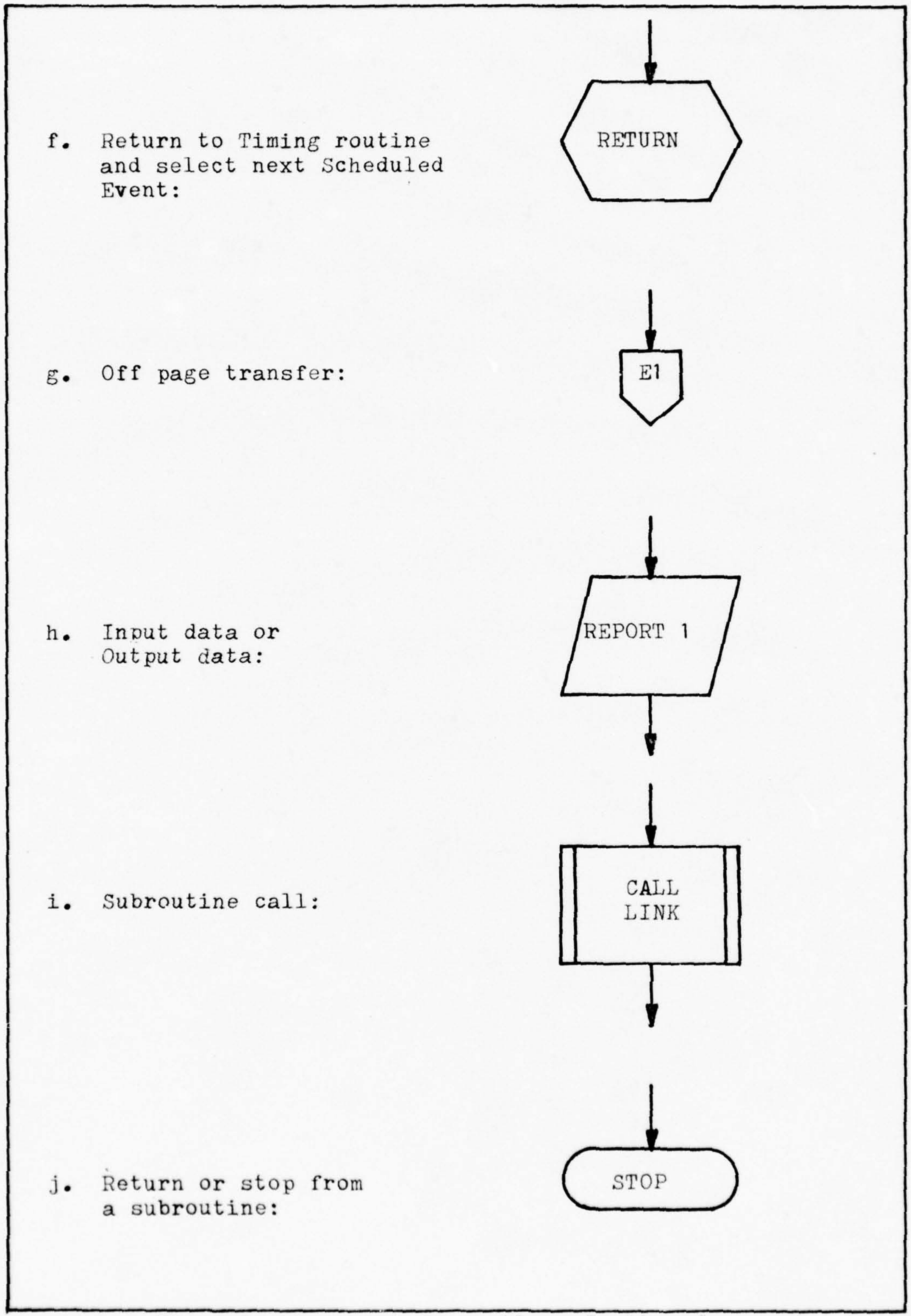


Fig A-1. Flowchart Conventions



f. Return to Timing routine and select next Scheduled Event:

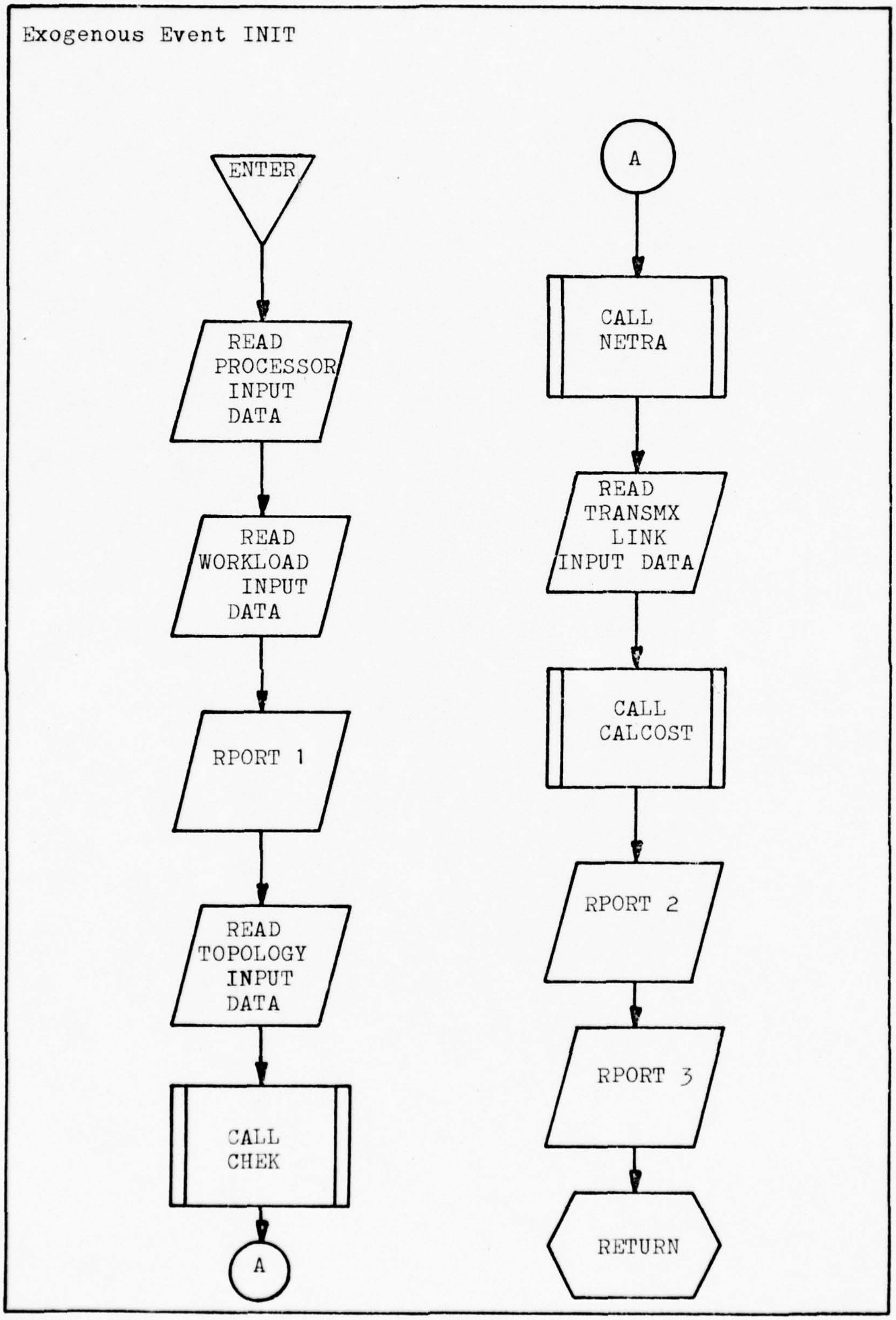
g. Off page transfer:

h. Input data or Output data:

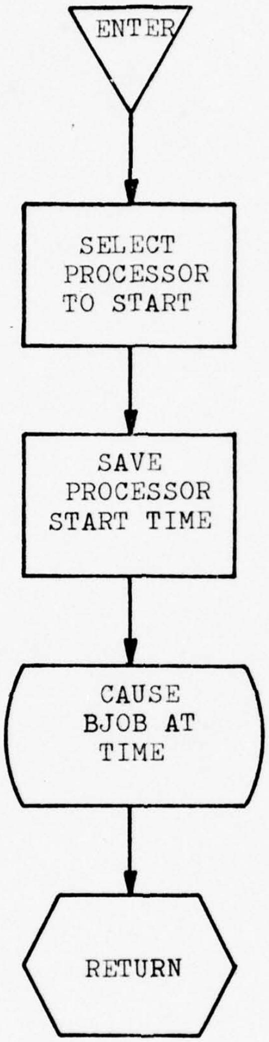
i. Subroutine call:

j. Return or stop from a subroutine:

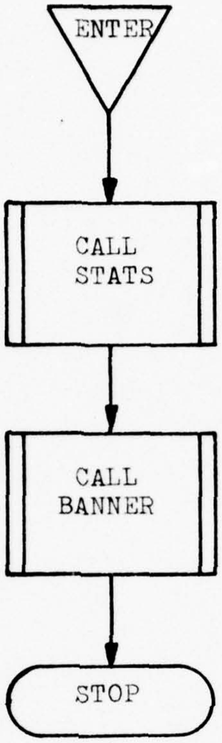
Fig A-1. Flowchart Conventions Continued


Exogenous Event INIT
ENTER
READ PROCESSOR INPUT DATA
READ WORKLOAD INPUT DATA
RPORT 1
READ TOPOLOGY INPUT DATA
CALL CHEK
A
A
CALL NETRA
READ TRANSMX LINK INPUT DATA
CALL CALCOST
RPORT 2
RPORT 3
RETURN

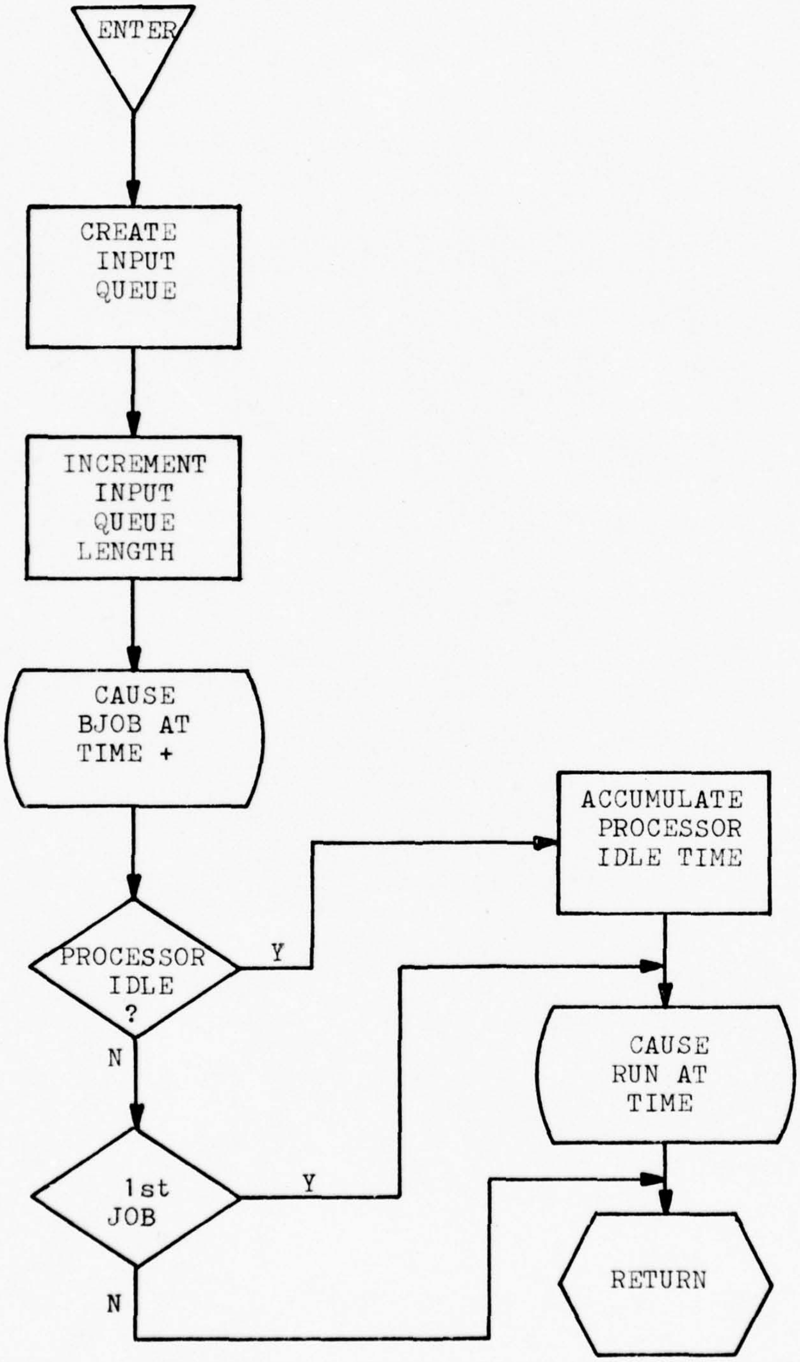
Exogenous Event START

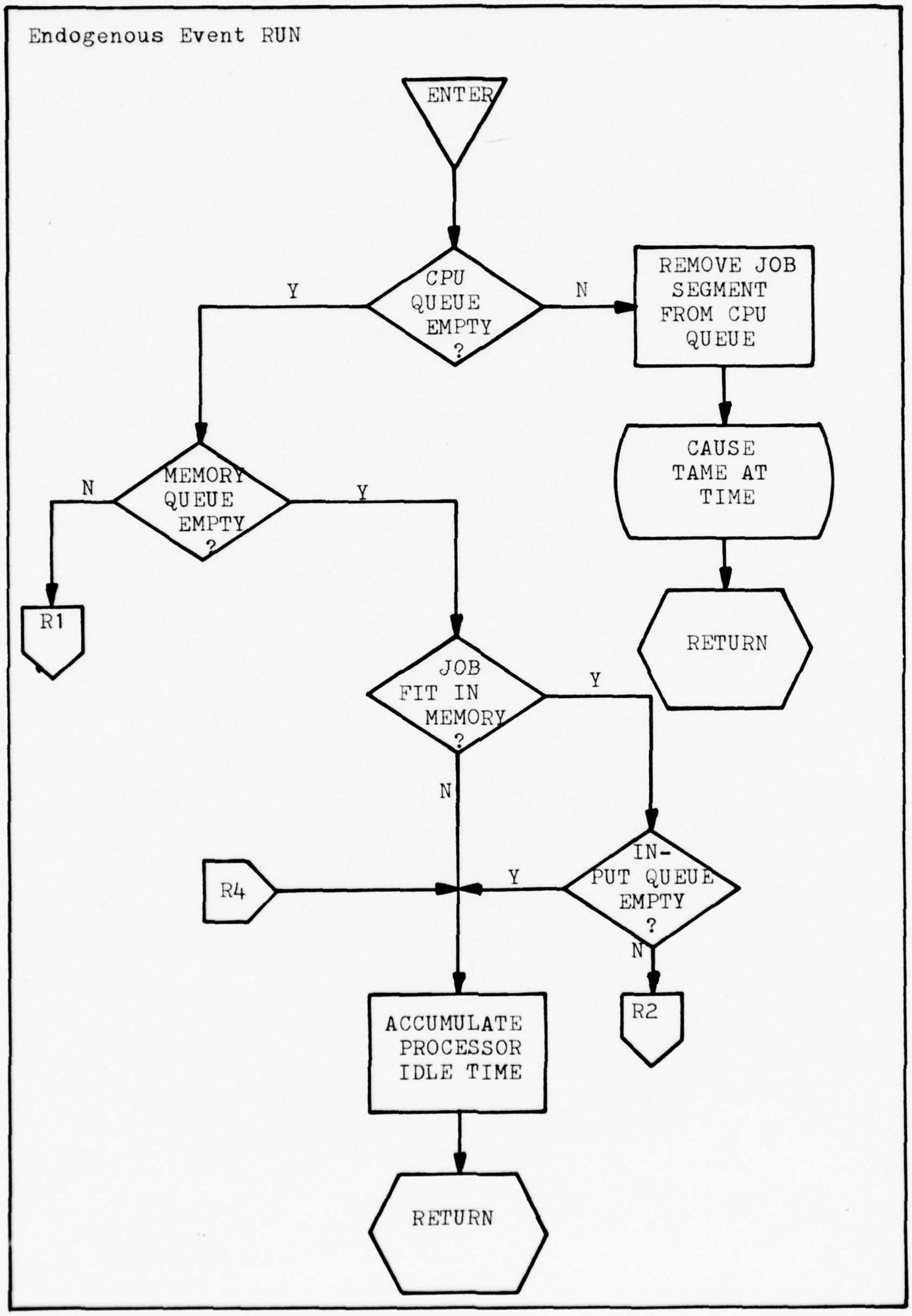


Exogenous Event END

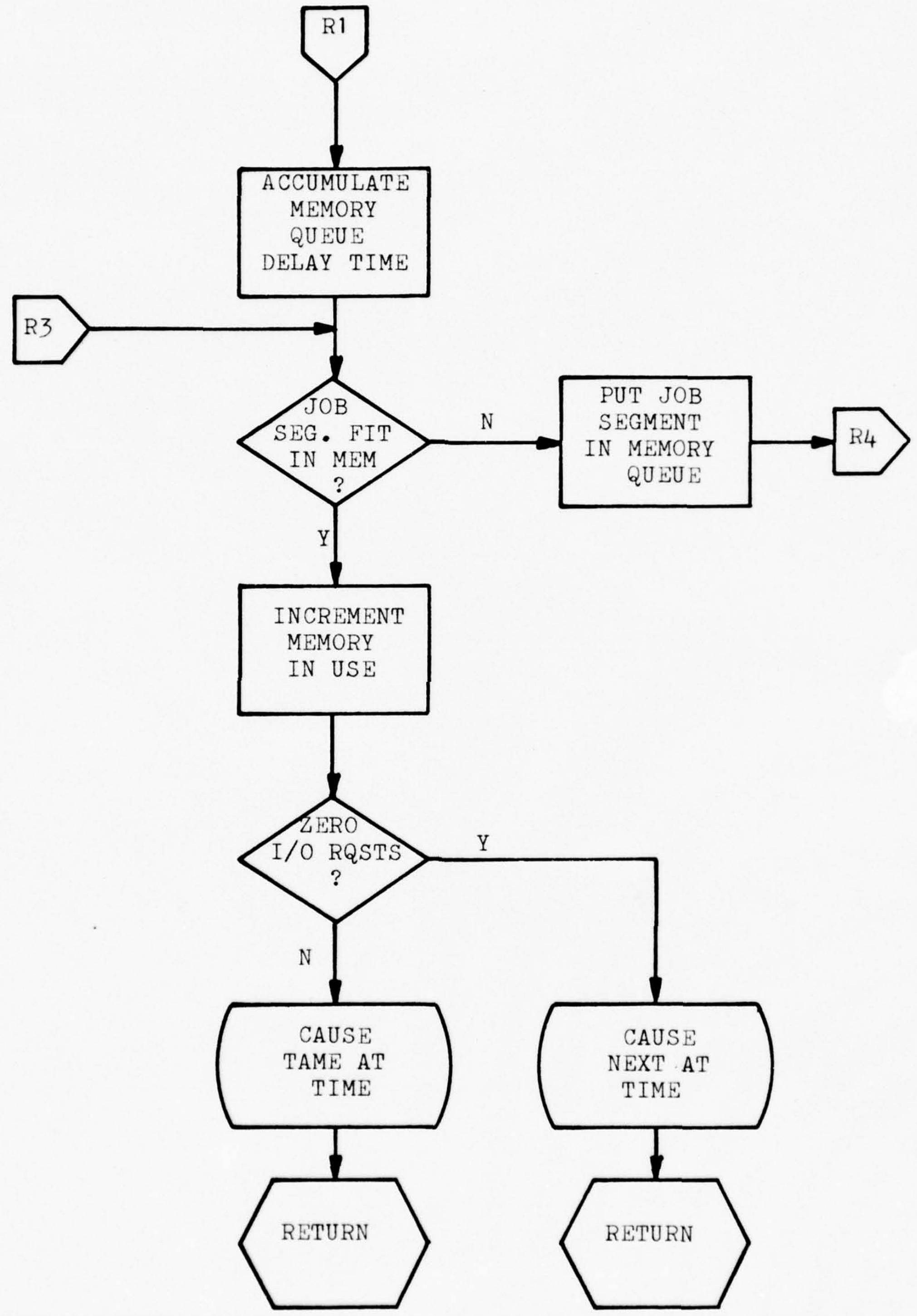


Endogenous Event BJOB

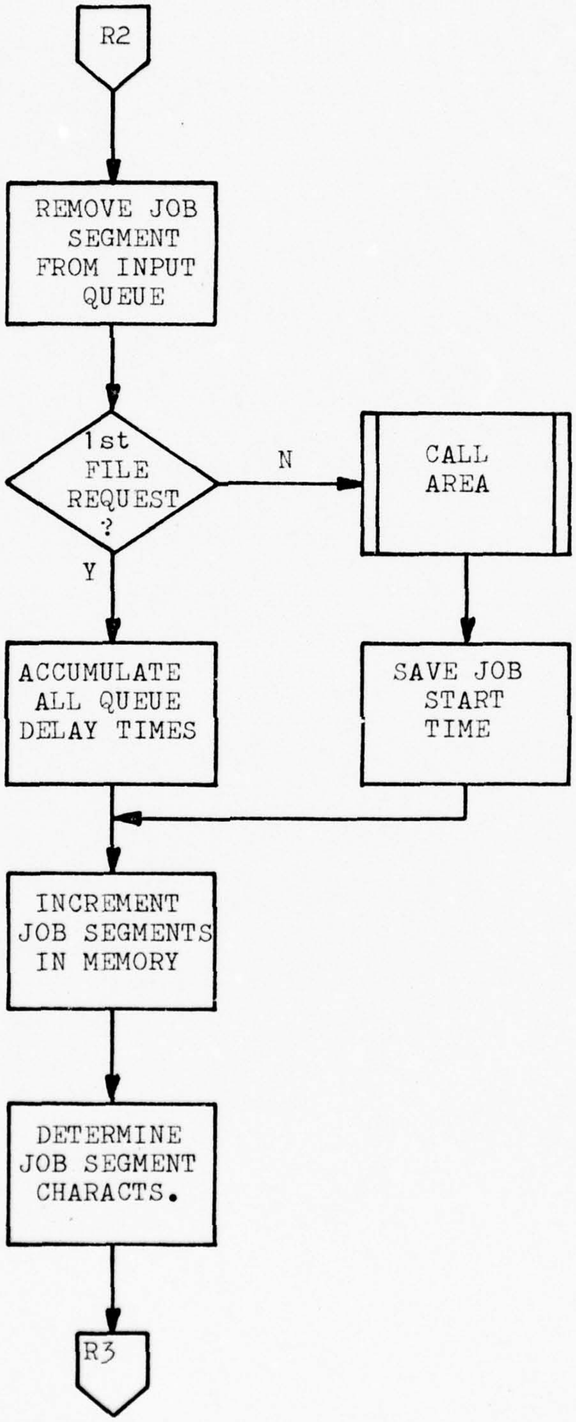



Endogenous Event RUN
ENTER
CPU QUEUE EMPTY ?
Y
N
REMOVE JOB SEGMENT FROM CPU QUEUE
CAUSE TAME AT TIME
RETURN
MEMORY QUEUE EMPTY ?
N
Y
R1
JOB FIT IN MEMORY ?
Y
N
R4
Y
IN-PUT QUEUE EMPTY ?
N
R2
ACCUMULATE PROCESSOR IDLE TIME
RETURN

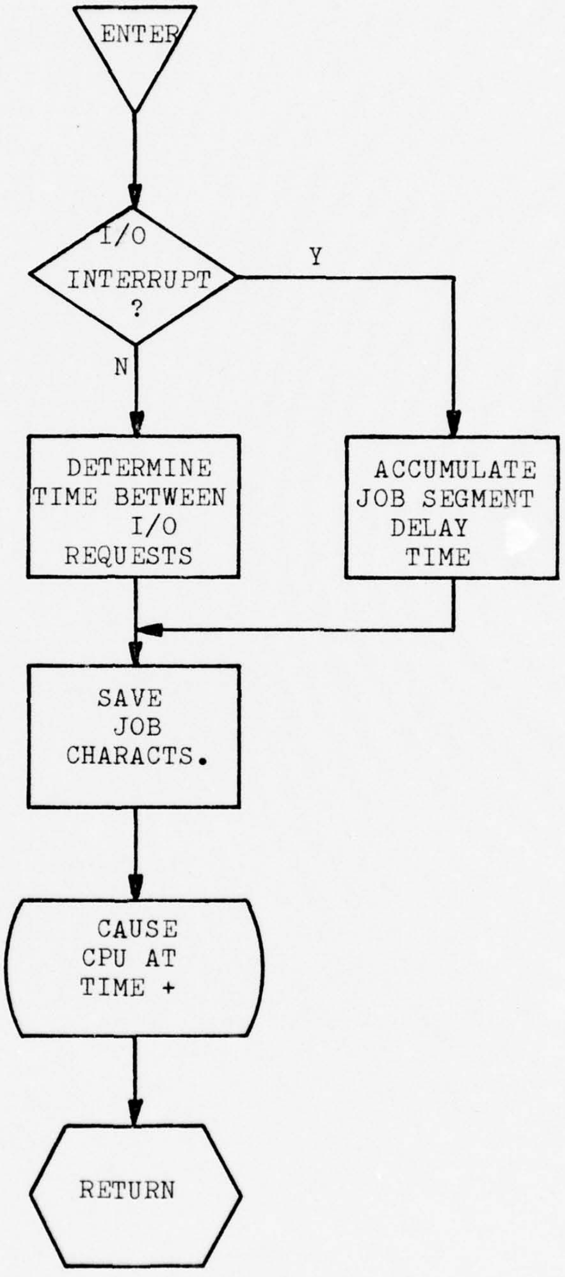
Endogenous Event RUN (Cont.)



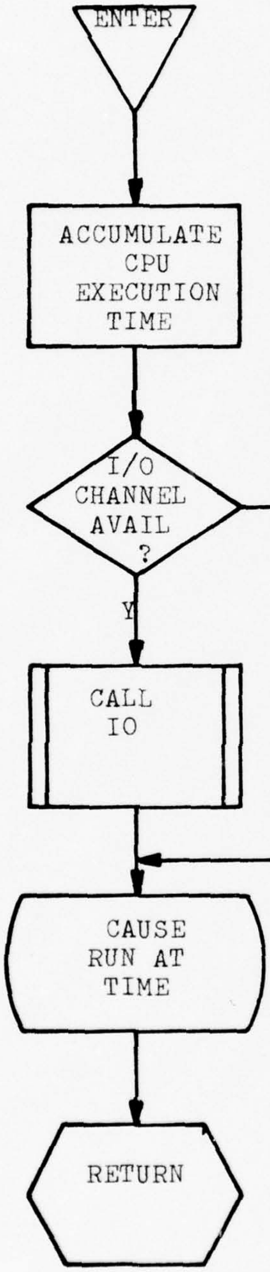
Endogenous Event RUN (cont.)


R2
REMOVE JOB SEGMENT FROM INPUT QUEUE
1st FILE REQUEST ?
N
CALL AREA
Y
ACCUMULATE ALL QUEUE DELAY TIMES
SAVE JOB START TIME
INCREMENT JOB SEGMENTS IN MEMORY
DETERMINE JOB SEGMENT CHARACTS.
R3

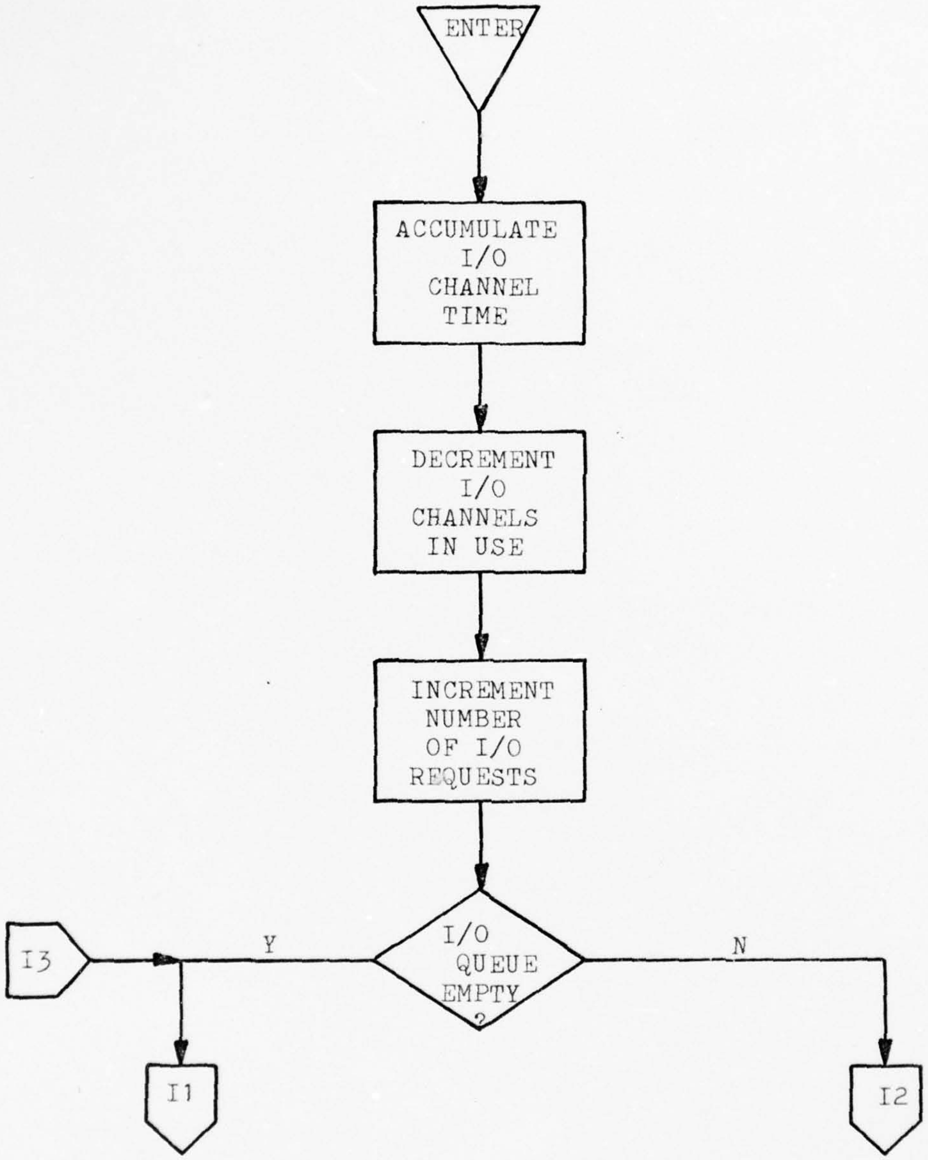
Endogenous Event TAME


ENTER
I/O INTERRUPT ?
Y
N
DETERMINE TIME BETWEEN I/O REQUESTS
ACCUMULATE JOB SEGMENT DELAY TIME
SAVE JOB CHARACTS.
CAUSE CPU AT TIME +
RETURN

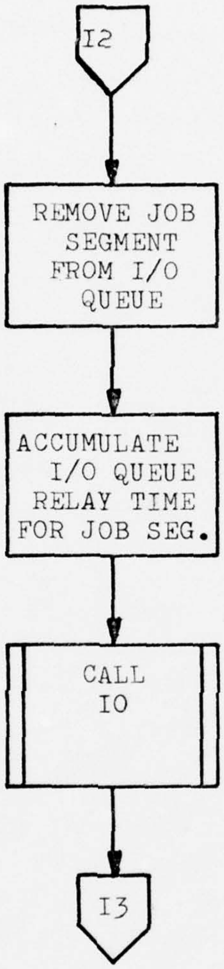
Endogenous Event CPU



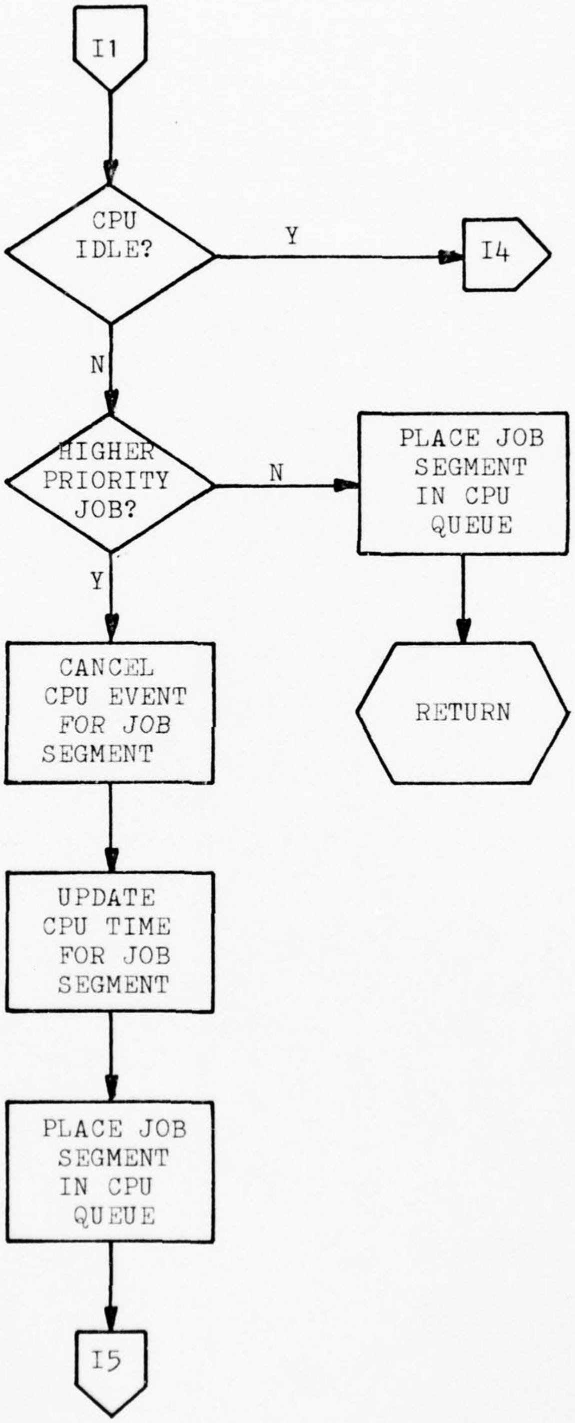
Endogenous Event INTER


ENTER
ACCUMULATE I/O CHANNEL TIME
DECREMENT I/O CHANNELS IN USE
INCREMENT NUMBER OF I/O REQUESTS
I/O QUEUE EMPTY ?
Y
N
I3
I1
I2

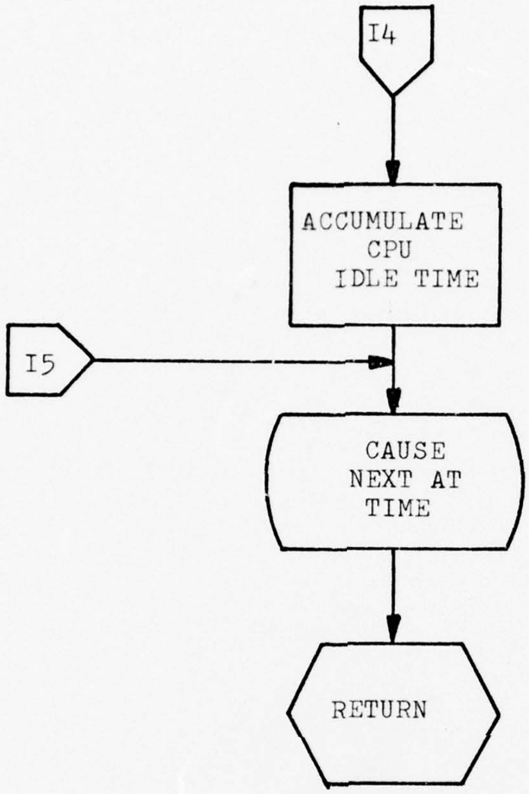
Endogenous Event INTER (Cont.)


I2
REMOVE JOB SEGMENT FROM I/O QUEUE
ACCUMULATE I/O QUEUE RELAY TIME FOR JOB SEG.
CALL IO
I3

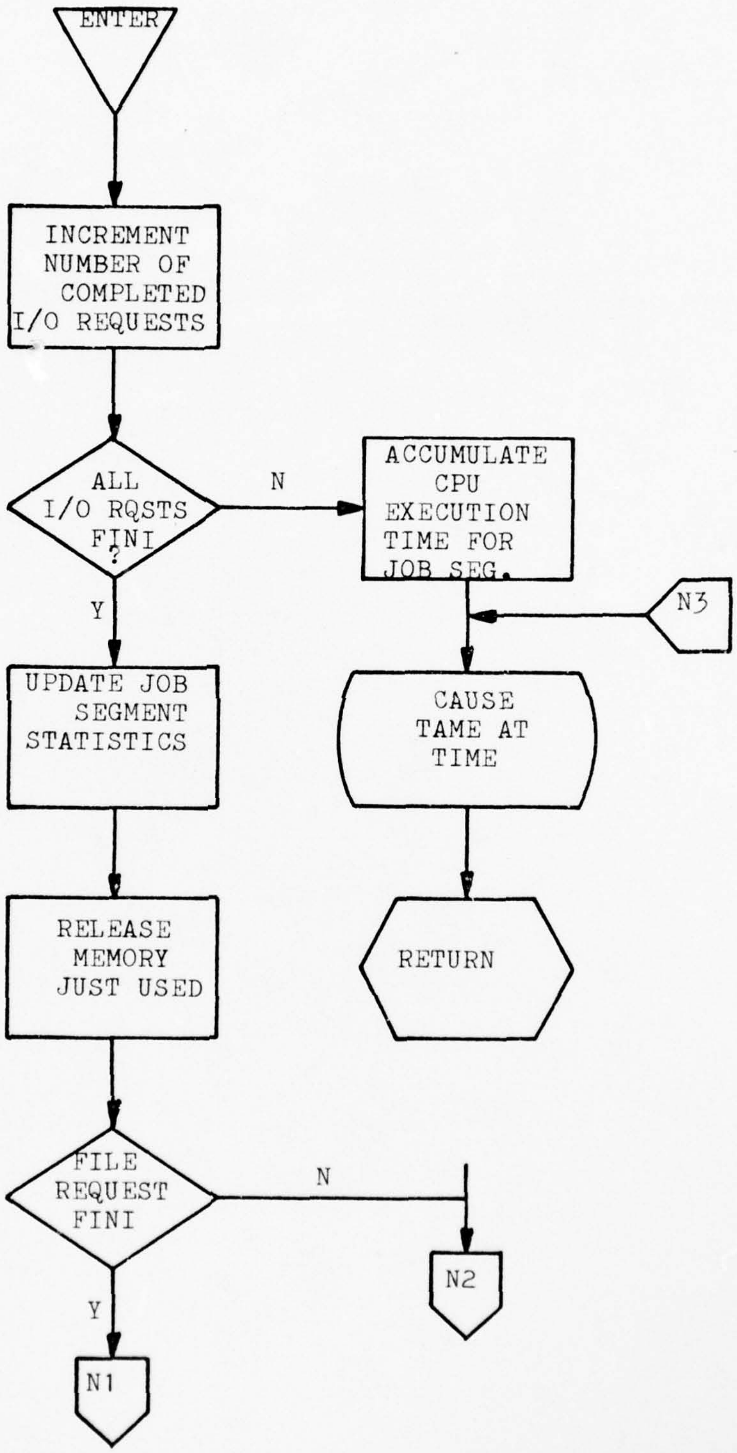
Endogenous Event INTER (Cont.)


I1
CPU IDLE?
Y
I4
N
HIGHER PRIORITY JOB?
N
PLACE JOB SEGMENT IN CPU QUEUE
RETURN
Y
CANCEL CPU EVENT FOR JOB SEGMENT
UPDATE CPU TIME FOR JOB SEGMENT
PLACE JOB SEGMENT IN CPU QUEUE
I5

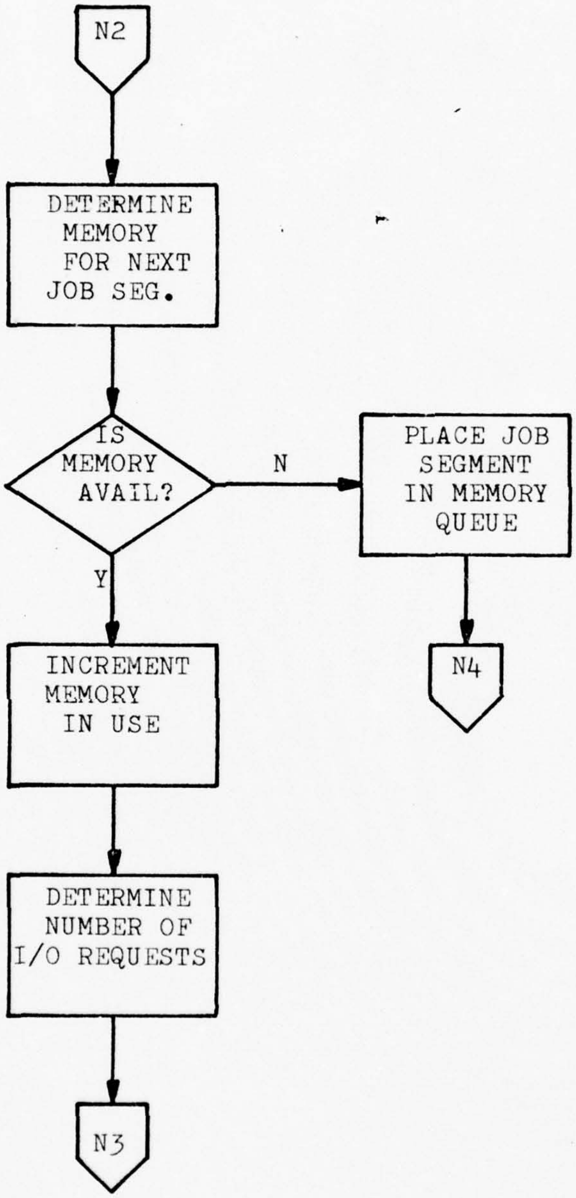
Endogenous Event INTER (Cont.)



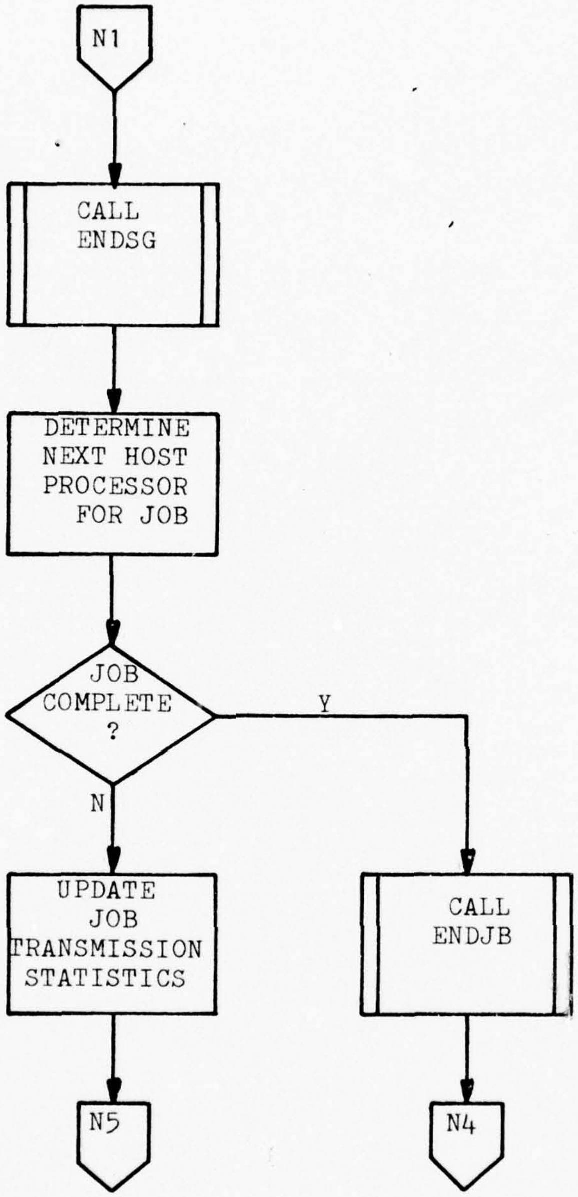
Endogenous Event NEXT


ENTER
INCREMENT NUMBER OF COMPLETED I/O REQUESTS
ALL I/O RQSTS FINI ?
N
Y
ACCUMULATE CPU EXECUTION TIME FOR JOB SEG.
N3
CAUSE TAME AT TIME
RETURN
UPDATE JOB SEGMENT STATISTICS
RELEASE MEMORY JUST USED
FILE REQUEST FINI
N
Y
N2
N1

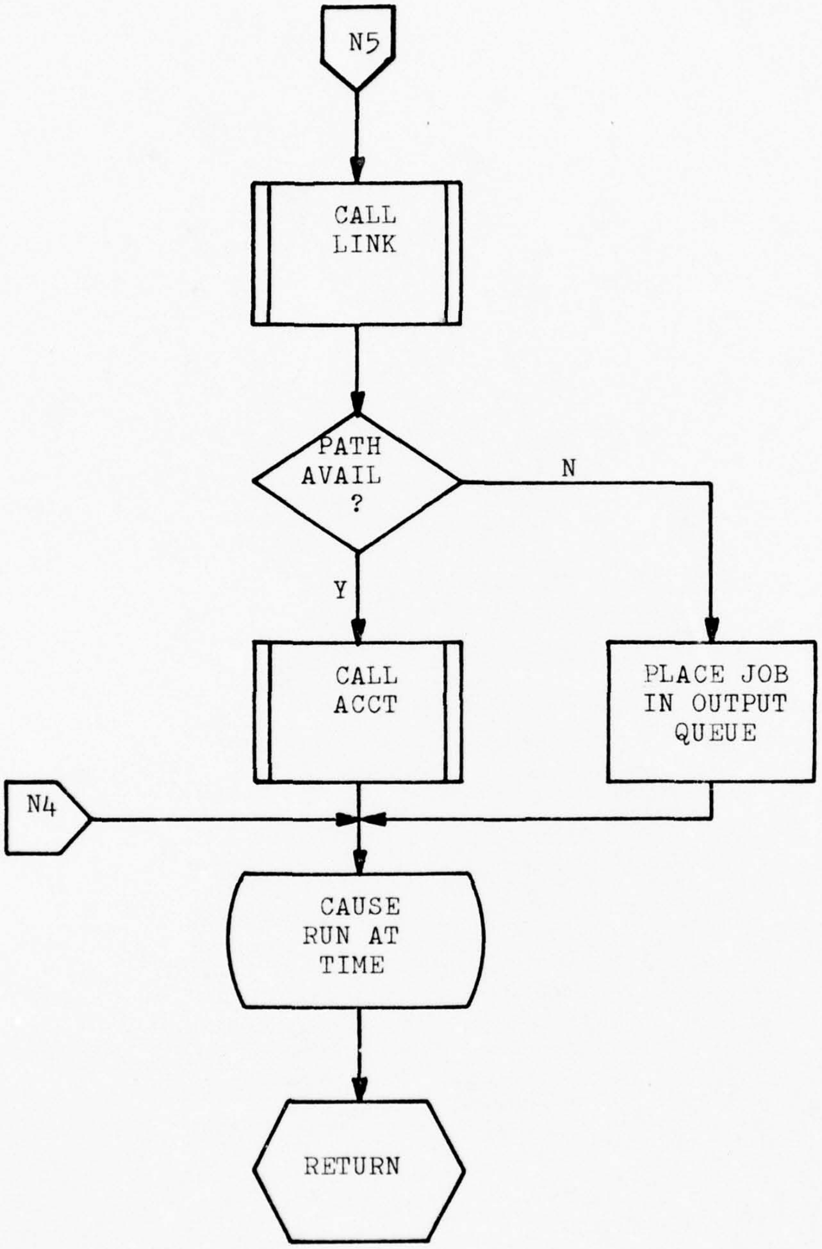
Endogenous Event NEXT (Cont.)

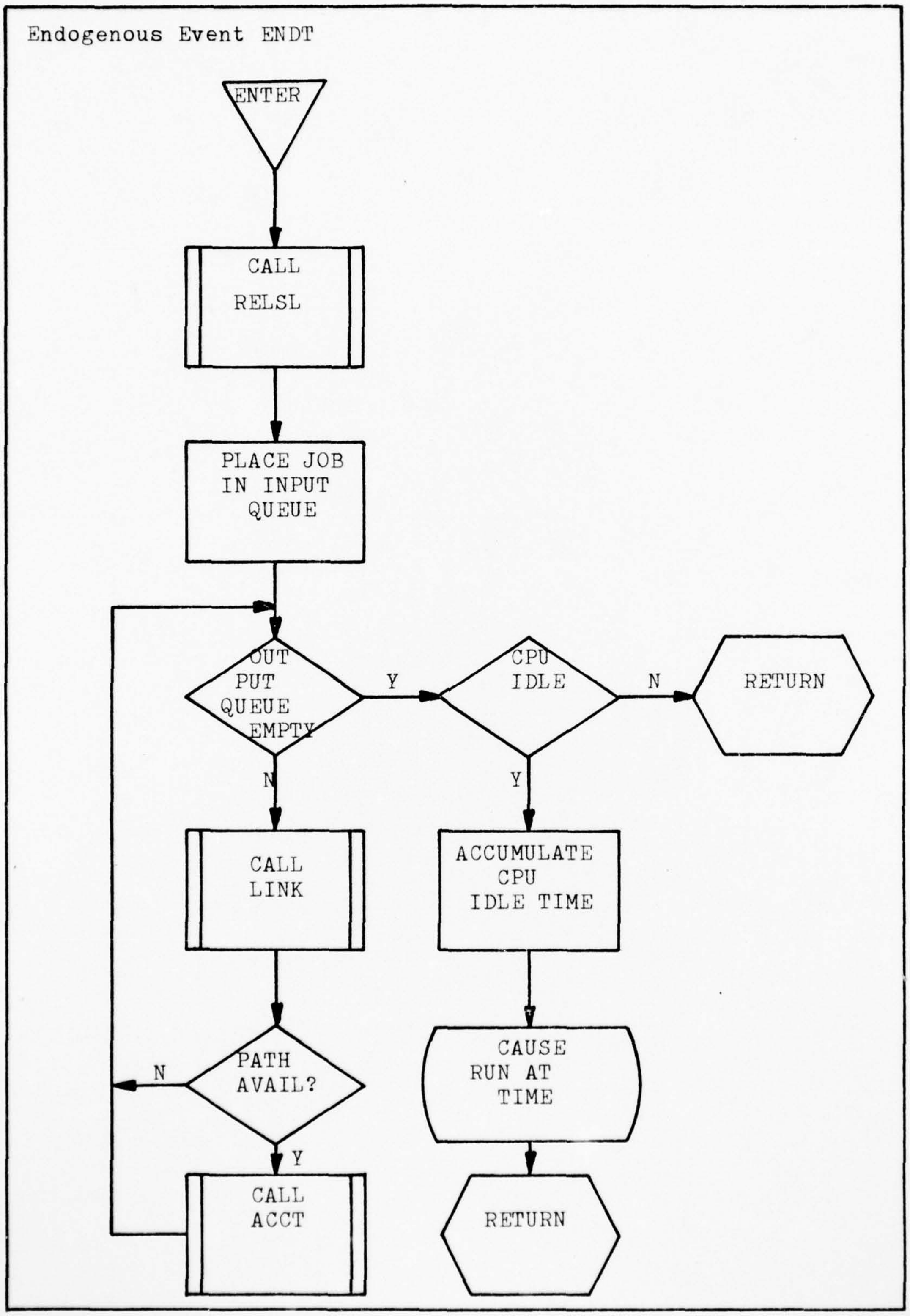

N2
DETERMINE MEMORY FOR NEXT JOB SEG.
IS MEMORY AVAIL?
N
PLACE JOB SEGMENT IN MEMORY QUEUE
N4
Y
INCREMENT MEMORY IN USE
DETERMINE NUMBER OF I/O REQUESTS
N3

Endogenous Event NEXT (Cont.)

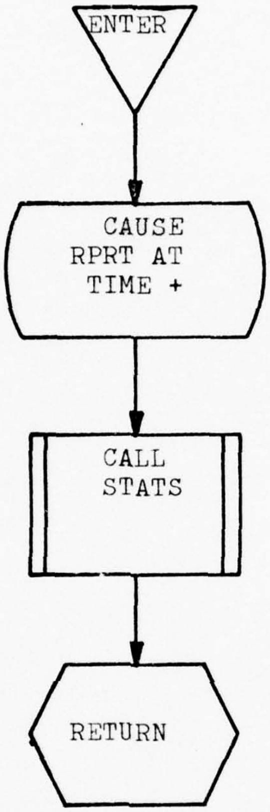

N1
CALL ENDSG
DETERMINE NEXT HOST PROCESSOR FOR JOB
JOB COMPLETE ?
Y
N
UPDATE JOB TRANSMISSION STATISTICS
CALL ENDJB
N5
N4

Endogenous Event NEXT (Cont.)



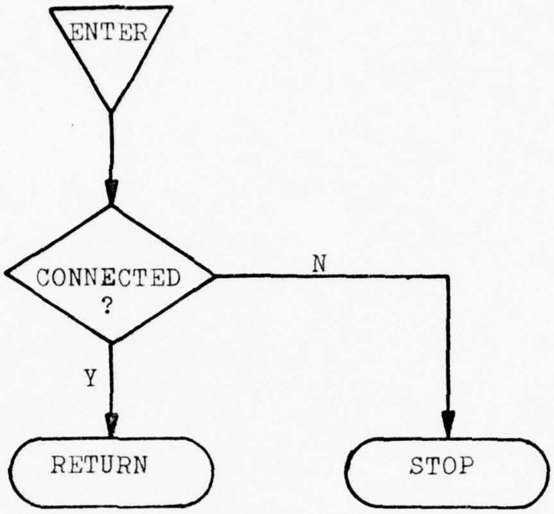

Endogenous Event ENDT
ENTER
CALL
RELSL
PLACE JOB
IN INPUT
QUEUE
OUT
PUT
QUEUE
EMPTY
Y
N
CPU
IDLE
N
RETURN
Y
CALL
LINK
ACCUMULATE
CPU
IDLE TIME
N
PATH
AVAIL?
Y
CAUSE
RUN AT
TIME
CALL
ACCT
RETURN

Endogenous Event RPRT

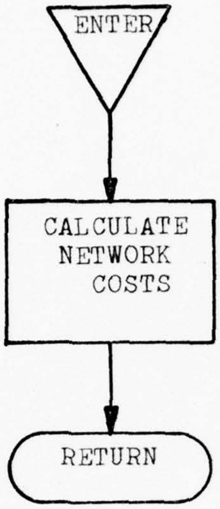


SIMSCRIPT I.5

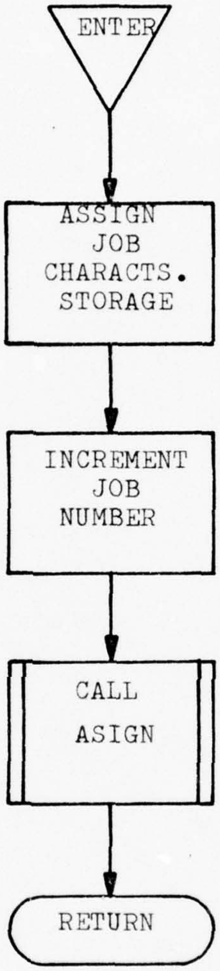
Subroutine CHEK



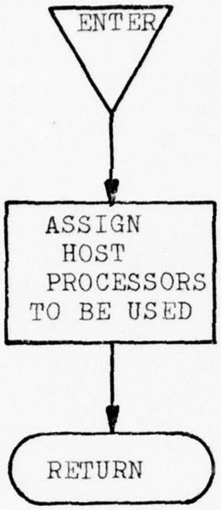
SIMSCRIPT I.5
Subroutine CALCOST



SIMSCRIPT I.5
Subroutine AREA

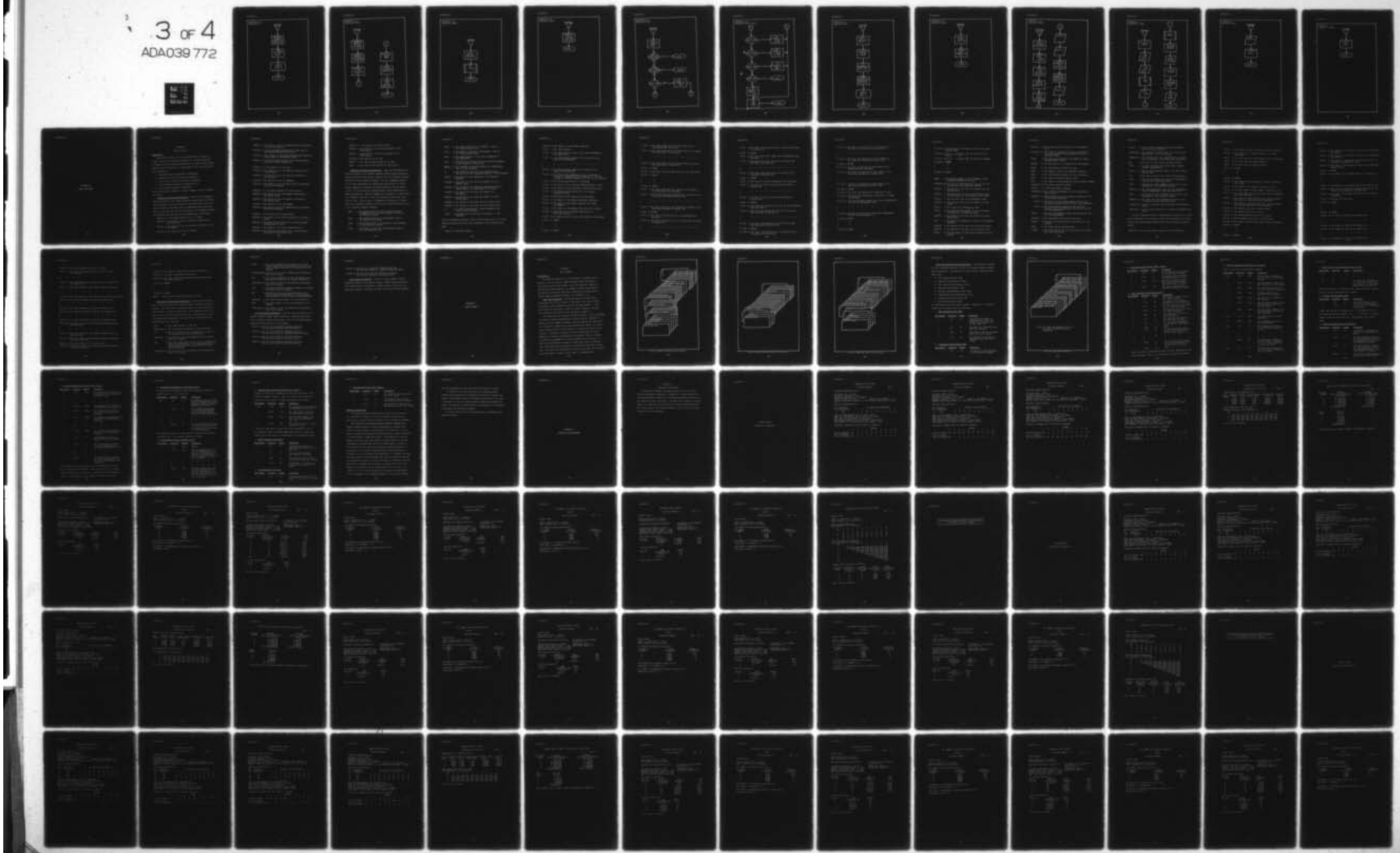


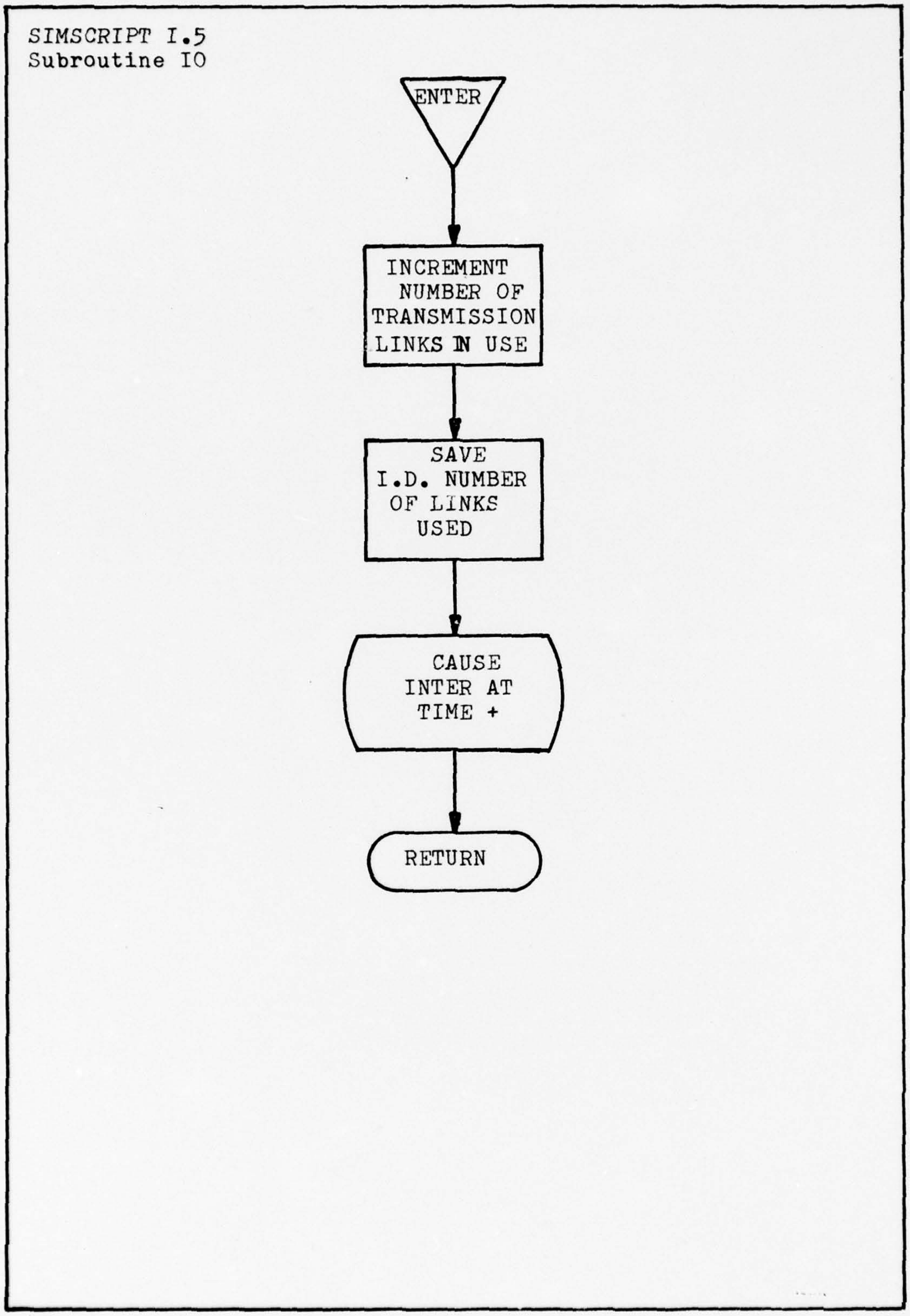
SIMSCRIPT I.5
Subroutine ASIGN



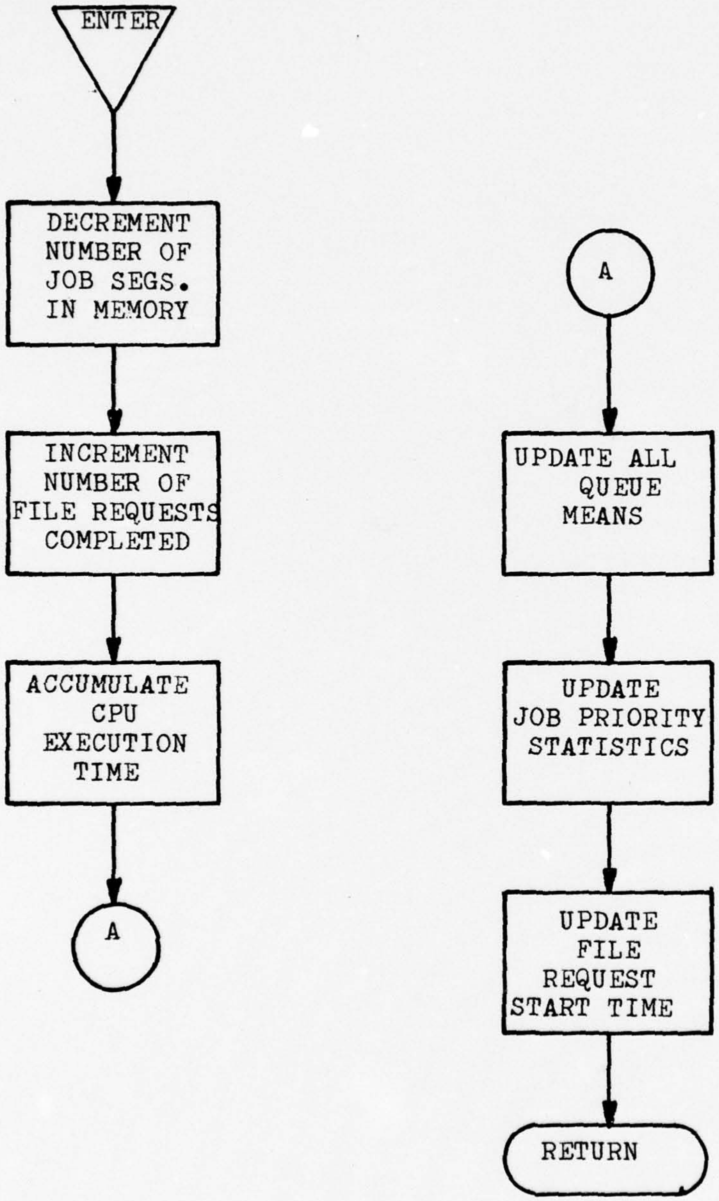
AD-A039 772 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 9/2
A GENERAL COMPUTER NETWORK SIMULATION MODEL.(U)
MAR 77 H M WARREN
UNCLASSIFIED AFIT/GCS/EE/77-3 NL

3 OF 4
ADA039772

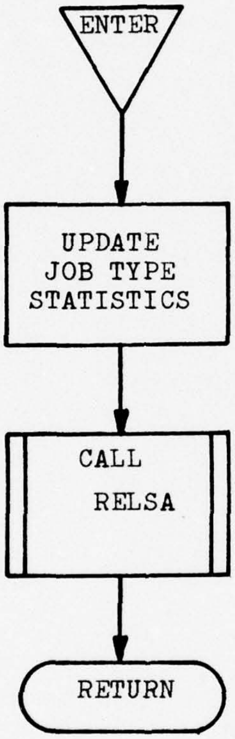



SIMSCRIPT I.5
Subroutine IO
ENTER
INCREMENT NUMBER OF TRANSMISSION LINKS IN USE
SAVE I.D. NUMBER OF LINKS USED
CAUSE INTER AT TIME +
RETURN

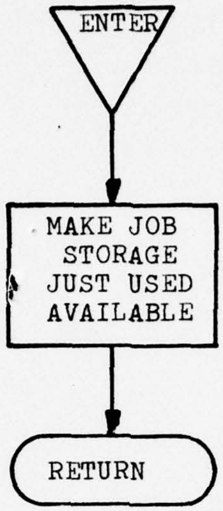
SIMSCRIPT I.5
Subroutine ENDSG

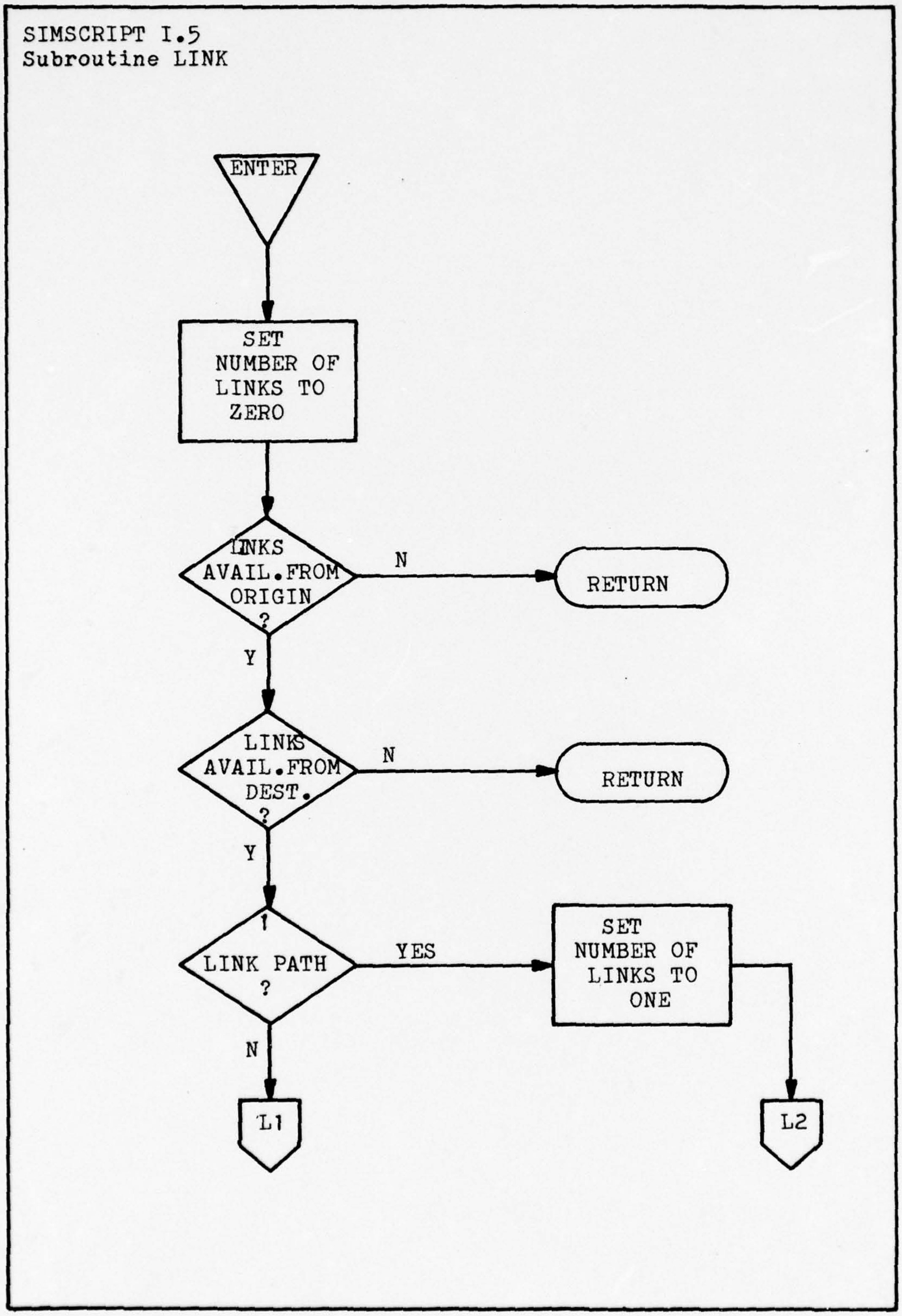

ENTER
DECREMENT NUMBER OF JOB SEGS. IN MEMORY
INCREMENT NUMBER OF FILE REQUESTS COMPLETED
ACCUMULATE CPU EXECUTION TIME
A
A
UPDATE ALL QUEUE MEANS
UPDATE JOB PRIORITY STATISTICS
UPDATE FILE REQUEST START TIME
RETURN

SIMSCRIPT I.5
Subroutine ENDJB

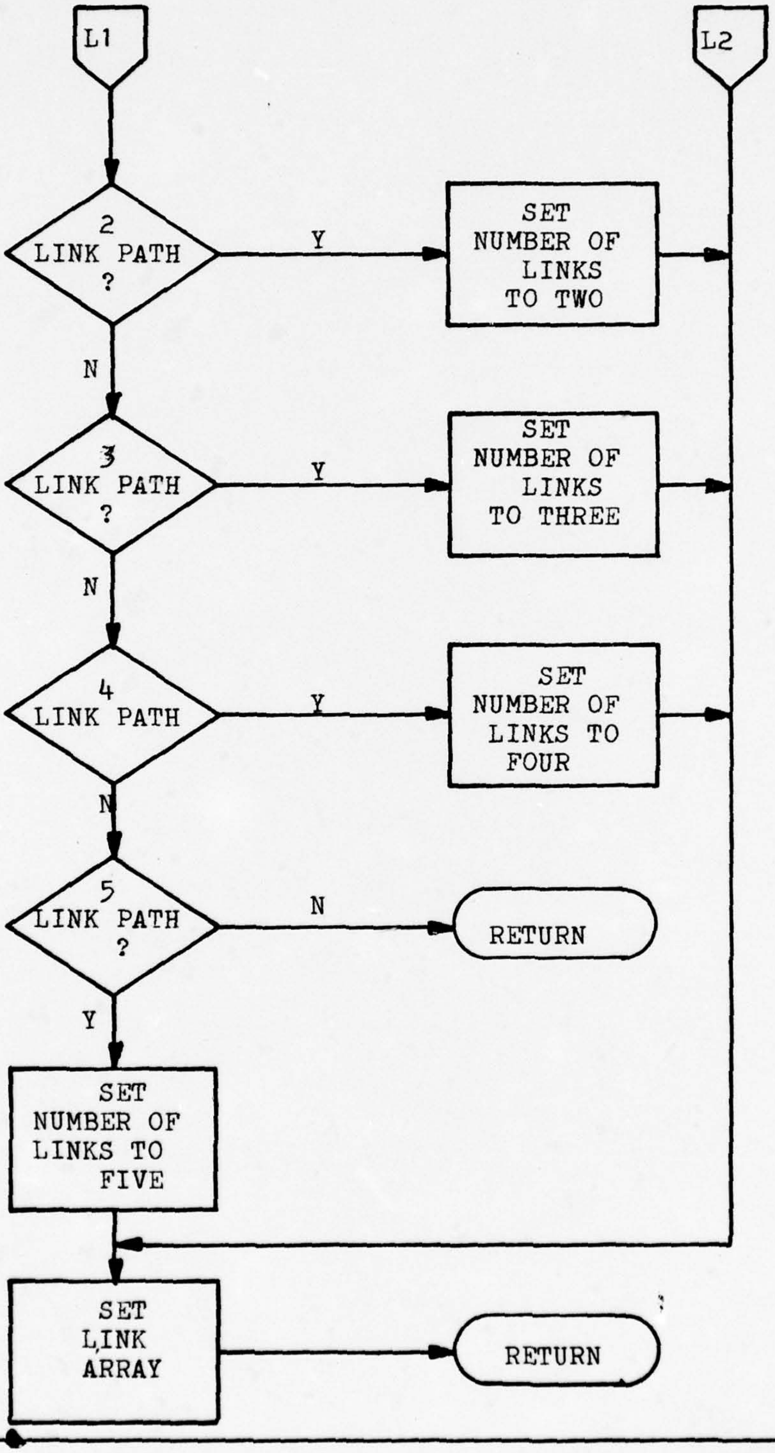


SIMSCRIPT I.5
Subroutine RELSA

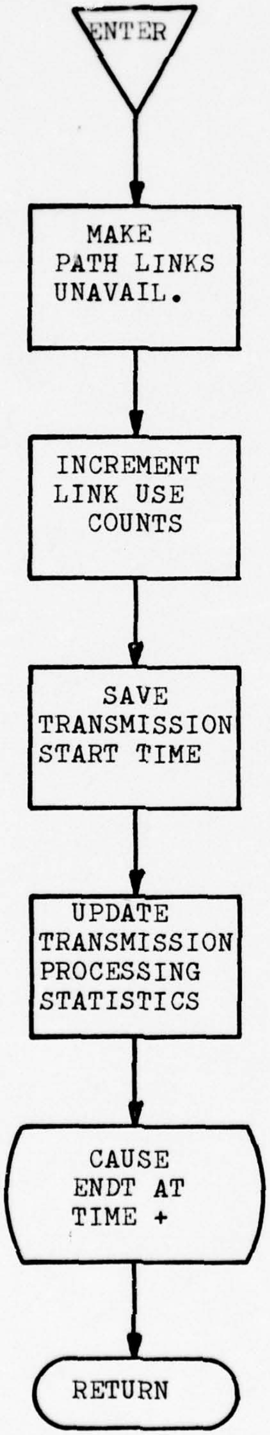



SIMSCRIPT I.5
Subroutine LINK
ENTER
SET NUMBER OF LINKS TO ZERO
LINKS AVAIL.FROM ORIGIN ?
N
RETURN
Y
LINKS AVAIL.FROM DEST. ?
N
RETURN
Y
LINK PATH ?
YES
SET NUMBER OF LINKS TO ONE
N
L1
L2

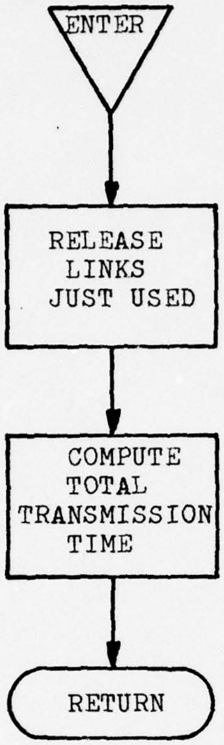
SIMSCRIPT I.5
Subroutine LINK (Cont.)

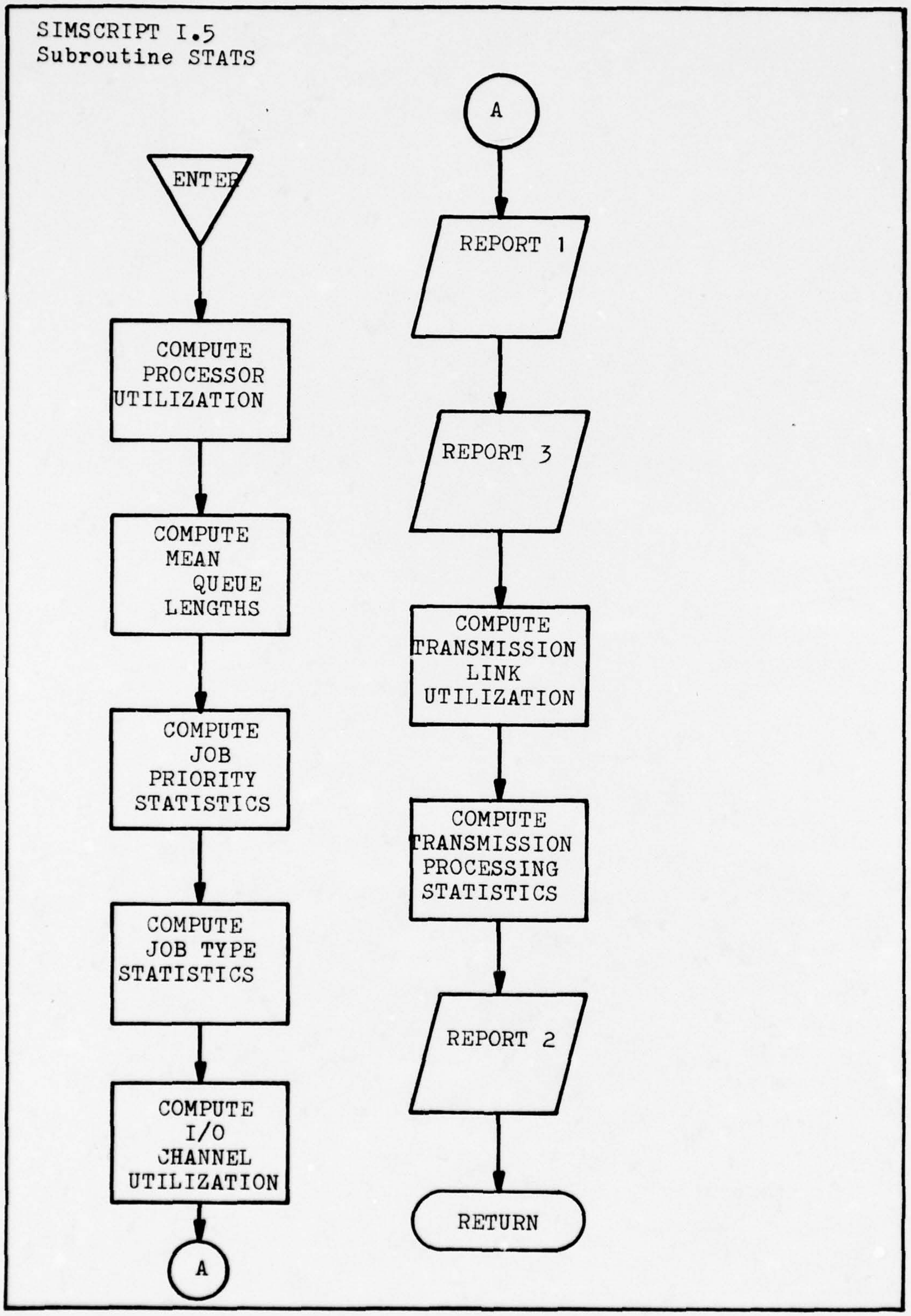

L1
L2
2 LINK PATH ?
Y
SET NUMBER OF LINKS TO TWO
N
3 LINK PATH ?
Y
SET NUMBER OF LINKS TO THREE
N
4 LINK PATH
Y
SET NUMBER OF LINKS TO FOUR
N
5 LINK PATH ?
N
RETURN
Y
SET NUMBER OF LINKS TO FIVE
SET LINK ARRAY
RETURN

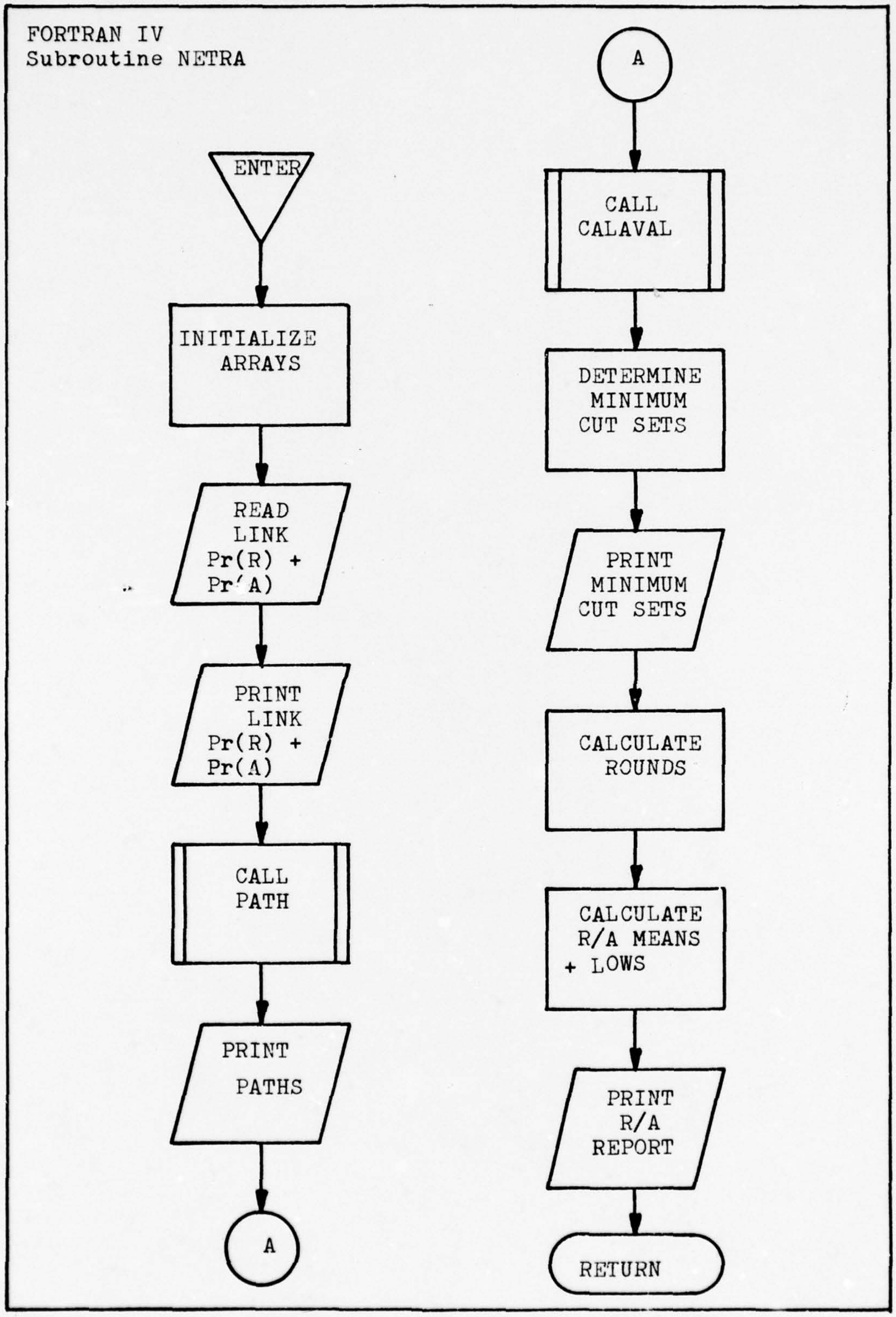
SIMSCRIPT I.5
Subroutine ACCT


ENTER
MAKE PATH LINKS UNAVAIL.
INCREMENT LINK USE COUNTS
SAVE TRANSMISSION START TIME
UPDATE TRANSMISSION PROCESSING STATISTICS
CAUSE ENDT AT TIME +
RETURN

SIMSCRIPT I.5
Subroutine RELSL

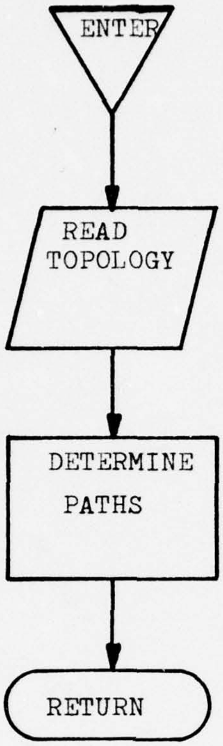



SIMSCRIPT I.5
Subroutine STATS
ENTER
COMPUTE PROCESSOR UTILIZATION
COMPUTE MEAN QUEUE LENGTHS
COMPUTE JOB PRIORITY STATISTICS
COMPUTE JOB TYPE STATISTICS
COMPUTE I/O CHANNEL UTILIZATION
A
A
REPORT 1
REPORT 3
COMPUTE TRANSMISSION LINK UTILIZATION
COMPUTE TRANSMISSION PROCESSING STATISTICS
REPORT 2
RETURN

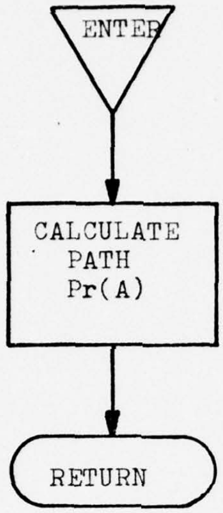


FORTRAN IV
Subroutine NETRA

FORTRAN IV
Subroutine PATH



FORTRAN IV
Subroutine CALAVAL



APPENDIX B

DATA STRUCTURE

APPENDIX B

Data Structure

Introduction

This section defines the data structure used to support the general computer network system simulation. The data structure is divided into five segments according to the type of state information stored or maintained in each segment. The five data structure segments are:

1. Temporary Entity and Attribute information,
2. Permanent Entity and Attribute information,
3. Event Notice Entity and Attribute information,
4. Set and Ranking Attribute information, and
5. Local Routine Variable information.

Each data structure segment is defined separately and in alphabetical order.

Temporary Entities and Attributes. There are six Temporary Entities defined in the general computer network system simulation. Each Temporary Entity has from two to six Attributes. The N subscript indicates that there can be a Temporary Entity and Attribute of this type for each host processor in the simulated network. The Temporary Entities and their Attributes are the following:

TCHLQ(N) — A job segment waiting in an I/O queue.

PCHLQ(N) — The pointer to the job segment's predecessor in the I/O queue.

SCHLQ(N) — The pointer to the job segment's successor in the I/O queue.

PRIO3(N) — The priority of the job segment.

JPNT3(N) — The pointer to the job characteristics associated with the job segment.

TCONC(N) — A host processor identification number used in determining network connectivity.

SCONC(N) — The pointer to succeeding host processor identification number in connectivity stack.

NCONC(N) — The identification number of the host processor connected to host processor N.

TCPUQ(N) — A job segment in a CPU queue.

PCPUQ(N) — The pointer to the job segment's predecessor in the CPU queue.

SCPUQ(N) — The pointer to the job segment's successor in the CPU queue.

PRIO2(N) — The priority of the job segment.

JPNT2(N) — The pointer to the job characteristics associated with the job segment.

TMEMQ(N) — A job segment waiting in a memory queue.

PMEMQ(N) — The pointer to the job segment's predecessor in the memory queue.

SMEMQ(N) — The pointer to the job segment's successor in the memory queue.

PRIO4(N) — The priority of the job segment.

JPNT4(N) — The pointer to the job characteristics associated with the job segment.

TOUTQ(N) — A job waiting in an output queue.

SOUTQ(N) — The pointer to the job's successor in the output queue.

NODE5(N) — The identification number of the host processor transmitting the job.

JPNT5(N) — The pointer to the job's characteristics.

NXTP5(N) — The identification number of the host processor receiving the transmitted job.

TQUEU(N) — A job waiting in an input queue.

PQUEU(N) — The pointer to the job's predecessor in the input queue.

SQUEU(N) — The pointer to the job's successor in the input queue.

PRIO1(N) — The priority of the job.

JNUM(N) — The identification number of the job.

QT(N) — The time the job entered the input queue.

Permanent Entities and Attributes. There are 68 Permanent Attributes defined in the general computer network system simulation. These 68 Attributes are used to describe the five Permanent Entities specified in the simulation program. The five Permanent Entities are the host processors, the transmission facilities, the transmission management software, the system workload, and the System itself. The attributes and nonsubscripted, singly subscripted, or doubly subscripted. When an N appears as a subscript, it means there is a Permanent Attribute of this type associated with each transmission link. Finally, when a J appears as a subscript, it means there is a Permanent Attribute of this type associated with each job in the simulated computer network.

ARYL1 — The maximum number of host processors (nodes) allowable in the simulated network. Used in the Initialization Deck.

ARYL2 — The number of entries in PROA(N,ARYL). Used in the Initialization Deck.

ARYL3 — The maximum number of transmission links allowable in the simulated network.

ARYL4 — The number of entries in WORK(N,ARYL4). Used in the Initialization Deck.

ARYL5 — The number of entries in OUT(ARYL5). Used in the Initialization Deck.

ARYL6 — The number of entries in JOB(J,ARYL6). Used in the Initialization Deck.

ARYL7 — The maximum number of job types allowable at a host processor.

AT(N) — The mean job inter-arrival time at a host processor. A real variable in time units.

BANDW(L) — The bandwidth in kilobits/second of a transmission link. A real variable.

DI — The number of time units in a simulated day, a simulated hour, or a simulated minute. A real variable.

ERUN(I) — The number of jobs completed in the simulated network.

FCHLQ(N) — The pointer to the first job segment in an I/O channel queue.

FCONC(N) — The pointer to the first host processor identification number in a connectivity test stack.

FCPUQ(N) — The pointer to the first job segment in a CPU queue.

FMEMQ(N) — The pointer to the first job segment in a main memory queue.

FOUTQ(N) — The pointer to the first job in an output queue.

FQUEU(N) — The pointer to the first job in an input queue.

ID — A scaling factor used to convert simulated time units. A real variable.

IOT(N) — The mean time in units per I/O request. A real variable.

The next Permanent Attribute (JOB(J, ARYL6)) holds all of the information concerning the job;s being processed by the simulated network.

JOB(J,1) — The job's number.

JOB(J,2) — The number of jobs already completed.

"(J,3) — The job's priority.

"(J,4) — The identification number of the job's originating host processor.

"(J,5) — The identification number of the second file request's host processor.

•
•
•

" " " " " " " " " " " "

"(J,12) — The identification number of the ninth file request's host processor.

"(J,13) — The index to host processor in use, or waiting to be used (0- originating host processor, 1- 1st file request, 2- 2nd file request, ..., 8- 8th file request)

"(J,14) — The CPU interrupt flag (4 = interrupt).

"(J,15) — The pointer to the job type characteristics in JTYP.

"(J,16) — The pointer to host processor currently being used.

"(J,17) — The identification number of the I/O channel currently in use.

"(J,18) — The number of job segments to be processed.

"(J,19) — The number of job segments currently completed.

"(J,20) — The amount of main memory required for a given job segment.

"(J,21) — The number of I/O requests for a given job segment.

"(J,22) — The number of I/O requests completed for a given job segment.

"(J,23) — The amount of CPU execution time between I/O requests.

"(J,24) — VACANT

• "
• "
• "

"(J,29) — VACANT

"(J,30) — The total number of I/O requests made in the originating host processor.

"(J,31) — The total number of I/O requests made in the second host processor to process job.

.
.   " " " " " " " " " " " " " " " "
.

"(J,38) — The total number of I/O requests made in the ninth host processor to process job.

"(J,39) — VACANT

"(J,40) — The start time for items JOB(J,41) thru JOB(J,120).

"(J,41) — VACANT

.   "
.   "
.   "

"(J,50) — VACANT

"(J,51) — The total delay time once execution is started at the originating host processor.

"(J,52) — The total delay time once execution is started at the second host processor to process job.

.
.   " " " " " " " " " " " " " " " " " "
.

"(J,59) — The total delay time once execution is started at the ninth host processor to process job.

"(J,60) — VACANT

"(J,61) — The total CPU execution time in originating host processor.

"(J,62) — The total CPU execution time in second host processor to process job.

.
.   " " " " " " " " " " " " " " " " "
.

"(J,69) — The total CPU execution time in ninth host processor to process job.

"(J,70) — VACANT

"(J,71) — The total input queue delay time in originating host processor.

"(J,72) — The total input queue delay time in second host processor to process job.

.
.   " " " " " " " " " " " " " " " " "
.

"(J,79) — The total input queue delay time in ninth host processor to process job.

"(J,80) — VACANT

"(J,81) — The total I/O time at originating host processor.

"(J,82) — The total I/O time at second host processor to process job.

.
.   " " " " " " " " " " " " " " " "
.

"(J,89) — The total I/O time at ninth host processor to process job.

"(J,90) — VACANT

"(J,91) — The total time waiting for the CPU at the originating host processor.

"(J,92) — The total time waiting for the CPU at the second host processor to process job.

.
.   " " " " " " " " " " " " " " "" "
.

"(J,99) — The total time waiting for the CPU at the ninth host processor to process job.

"(J,100) — VACANT

"(J,101) — The total time waiting for an I/O channel at the originating host processor.

"(J,102) — The total time waiting for an I/O channel at
the second host processor to process job.

.
. " " " " " " " " " " " " "
.

"(J,109) — The total time waiting for an I/O channel at
the ninth host processor to process job.

"(J,110) — VACANT

"(J,111) — The total time waiting for main memory at the
originating host processor.

"(J,112) — The total time waiting for main memory at the
second host processor to process job.

.
. " " " " " " " " " " " "
.

"(J,119) — The total time waiting for main memory at the
ninth host processor to process job.

"(J,120) — VACANT

"(J,121) — The total time waiting for transmission links.

"(J,122) — The number of transmission links used in a transmission path.

"(J,123) — The identification number of the first transmission
link in the transmission path.

.
. " " " " " " " " "
.

"(J,127) — The identification number of the fifth transmission
link in the transmission path.

"(J,128) — VACANT

. "
. "
. "

"(J,132) — VACANT

"(J,133) — The beginning of the waiting time for the transmission links.

"(J,134) — The begin processing time for the first file request of a job.

"(J,135) — The begin processing time for each file request.

"(J,136) — VACANT

. "
. "
. "

"(J,202) — VACANT

JSTOR — The maximum number of jobs allowable in the simulated network at one time.

JTYP(ARYL7,N)-The job type characteristics for all the host processors in the simulated network.

LCHLQ(N) — The pointer to the last job segment in an I/O channel queue.

LCDNC(N) — The pointer to the last host processor identification number in a connectivity test stack.

LCOST(L) — The per month cost of a transmission link.

LCPUQ(N) — The pointer to the last job segment in a CPU queue.

LGTH(L) — The length of a transmission link.

LINK(I) — The identification number of a host processor traversed by a transmission path.

LMEMQ(N) — The pointer to the last job segment in a main memory queue.

LNKN — The number of transmission links in the simulated network.

LOUTQ(N) — The pointer to the last job in an output queue.

LQUEU(N) — The pointer to the last job in an input queue.

MEM(N) — The mean amount of main memory required by a job segment.

MILEC(L) — The cost per mile per month for a transmission link.

MSGS(N,N) — The number of transmission requests transmitted over a transmission link expressed as a host processor tuple.

NFLAG — The flag used to activate the PARAMS and PRIODS Reports (0=OFF, 1-ON).

NIOR(N) — The mean number of I/O requests per job segment.

NMJOB — The identification number of a job.

NPNT — The number of jobs in the simulated network.

NPRO — The identification number of a host processor.

NTIME — The current simulated time in units.

NUM — The number of host processors in the simulated network.

OUT(ARYL5) — The output statistics buffer.

PLOC(N) — A host processor's location.

PPROB(N,N) — The cumulative probabilities for job priorities from 1 to 10.

PROA(N,1) — The priority of the job currently executing on the CPU at a host processor.

" (N,2) — The pointer to job characteristics for the job currently executing on the CPU at a host processor.

" (N,3) — The identification number for the CPU Event Notice for the job currently executing on the CPU at a host processor.

PROC(N) — The per month hardware rental and maintenance cost at a host processor.

PTYPE(N) — The nomenclature of the host processor.

PUTM(N) — The mean CPU time between I/O requests for a job segment.

RANDR — The random number generator root.

SIGN(N) — The stack pointer for the connectivity test stack for a host processor.

SOFC(N) — The per month software rental and maintenance cost at a host processor.

SPROB(N,N) — The cumulative probabilities for a file request having from 1 to 10 job segment.

STRAN(N,N) — The starting time for a transmission link being used to transmit a job between host processors.

TBR — The time in units between interval reports.

TCOST — The total per month cost for the basic computer network configuration.

TERSV(L) — The per month terminal service cost for a transmission link.

TFR — The time in units of the first interval report.

TLCOS — The total per month cost of all the transmission links in the simulated network.

TPROC — The total per month hardware cost of all the host processors in the simulated network.

TRANT(N,N) — The mean transmission time for each transmission link. Expressed as a host processor tuple.

TSOFC — The total per month software cost of all the host processors in the simulated network.

TTRAN(N,N) — The total time each transmission link is used. Expressed as a host processor tuple.

TUPLA(N,N) — The host processor tuples available as transmission links.

TUTIL(N,N) — The utilization factor for a particular transmission link.

The next Permanent Attribute (WORK(N,ARYL4)) holds all the information concerning the utilization of the host processors, the transmission links and workload characterized for the simulated computer network.

WORK(N,1) — The identification number of a host processor.

"(N,2) — The number of control points for a host processor.

"(N,3) — The number of I/O channel attached to a host processor.

"(N,4) — The total amount of main memory at a host processor.

"(N,5) — The start time for Report REPORT 1.

"(N,6) — The start time for Report REPORT 2.

"(N,7) — VACANT

. "
. "
. "

"(N,9) — VACANT

"(N,10) — The number of jobs currently in memory or waiting for memory.

"(N,11) — The number of I/O channels currently in use.

"(N,12) — The total amount of main memory currently in use.

"(N,13) — The total number of job types that can be originated at a host processor.

"(N,14) — The number of jobs processed with a host processor acting as the originating host processor.

"(N,15) — The number of jobs processed with a host processor acting as an additional host processor.

"(N,16) — The maximum input queue length allowed.

"(N,17) — The current input queue length.

"(N,18) — The maximum attained input queue length.

"(N,19) — The accumulated input queue length totals.

"(N,20) — The number of input queue length totals accumulated.

"(N,21) — VACANT

. "
. "
. "

"(N,24) — VACANT

"(N,25) — The number of job segments delayed due to unavailable main memory.

"(N,26) — The number of I/O requests delayed due to unavailable I/O channels.

"(N,27) — The number of transmission requests delayed due to unavailable transmission links.

"(N,28) — The number of job segments delayed due to unavailable control points.

"(N,29) — VACANT

"(N,30) — The number of I/O requests made on I/O channel one.

•
• " " " " " " " " " " "
•

"(N,39) — The number of I/O requests made on I/O channel ten.

"(N,40) — The total host processor utilization since the beginning of the report period times 100.

"(N,41) — The total CPU execution time for all file requests completed.

"(N,42) — The total CPU idle time.

"(N,43) — VACANT

• "
• "
• "

"(N,49) — VACANT

"(N,50) — The total I/O time for I/O channel one.

•
• " " " " " " " "
•

"(N,59) — The total I/O time for I/O channel ten.

"(N,60) — The start I/O time for I/O channel one.

• " " " " " " " "
•
•

"(N,69) — The start I/O time for I/O channel ten.

"(N,70) — The host processor idle flag (0 = idle).

"(N,71) — The total throughput time per job type one per file request.

• " " " " " " " "

•

•

"(N,90) — The total throughput time per job type nineteen per file request.

"(N,91) — The total number of jobs completed per job type one.

•

• " " " " " " " "

•

"(N,110) — The total number of jobs completed per job type nineteen.

"(N,111) — The total throughput time per job priority one.

• " " " " " " " "

•

•

"(N,120) — The total throughput time per job priority ten.

"(N,121) — The total CPU execution time per job priority one.

•

• " " " " " " " "

•

"(N,130) — The total CPU execution time per job priority ten.

"(N,131) — The total number of file requests completed per job priority one.

•

• " " " " " " " " "

•

"(N,140) — The total number of file requests completed per job priority ten.

"(N,141) — The total number of transmission requests originated by a host processor.

"(N,142) — The current length of the output queue.

"(N,143) — The total length of the output queue at the beginning of each file request.

"(N,144) — The maximum length attained by an output queue.

"(N,145) — The total transmission delay time.

"(N,146) — The total number of transmission requests initiated by a host processor.

"(N,147) — VACANT

. "

. "

. "

"(N,201) — VACANT

WRDSZ — The host processor's word size in bits.

Event Notice Entities and Attributes. There are eight Event Notice Entities defined in the general computer network system simulation. Each Event Notice Entity has from one to two Attributes. One Attribute is common to each Event. This Attribute is the time that the Event is next scheduled to occur. This Attribute is System assigned and maintained. The Event Notice Entities and their Attributes are the following:

BJOB — This Event initiates a new job.

MJOB(BJOB) — The identification number of a host processor.

CPU — This Event assigns the I/O channels to be used by the job segment and the schedules run.

JPNT7(CPU) — The pointer to the job characteristics associated with a job segment.

ENDT — This Event calls RELSL at the end of a transmission request and determines if the transmission links just made available are needed for other transmission requests.

JPNTA(ENDT) — The pointer to the job characteristics associated with a job segment.

INTER — This Event compares the priorities of the job segment presently being processed by the CPU and the job segment which has just completed its I/O request.

JPNT8(INTER)— The pointer to the job characteristic associated with a job segment.

NEXT — This Event determines the next job segment to be processed following an I/O request completion.

JPNT9(NEXT) — The pointer to the job characteristic associated with a job segment.

RPRT — This Event initiates snap-shot reports at intervals as specified by the analyst.

RUN — This Event selects the highest priority file request awaiting processing by a host processor and determines the file request's characteristics.

MRUN(RUN) — The identification number of a host processor.

TAME — This Event starts the CPU for processing a job segment.

JPNT6(TAME) — The pointer to the job characteristic associated with a job segment.

Sets and Ranking Attributes. There are six Sets defined in the general computer network system simulation. Associated with each Set is a Ranking scheme and Attribute. The Sets and their Ranking schemes and Attributes are the following:

CHLQ(N) — The Set of all TCHLQ(N) Temporary Entities. The Sets are ranked on high job priority.

CONC(N) — The Set of all TCONC(N) Temporary Entities. The Sets are ranked on a first-in-first-out basis.

CPUQ(N) — The Set of all TCPUQ(N) Temporary Entities. The Sets are ranked on high job priority.

MEMQ(N) — The Set of all TMEMQ(N) Temporary Entities. The Sets are ranked on high job priority.

OUTQ(N) — The Set of all TOUTQ(N) Temporary Entities. The Sets are ranked on a first-in-first-out basis.

QUEU(N) — The Set of all TQUEU(N) Temporary Entities. The Sets are ranked on high job priority.

Local Routine Variables. There are a large number of local variables defined in the general computer network system simulation. In all cases they are defined by a comment prior to their use in a particular routine. Therefore, they are not listed in this Appendix.

APPENDIX C

USER'S MANUAL

APPENDIX C

User's Manual

Introduction

This Appendix provides all the information necessary for an analyst to use the genetal computer network system simulation. This section covers three areas, input deck structure, data deck structure and specification, and computer resource requirements. The specifics of control card usage reference the requirements of Control Data Corporation's Version 3 of SIMSCRIPT I.5.

Input Deck Structure. The general computer network system simulation can be run in either of two modes. The choice of mode is dependent upon an analyst's desires and requirements. In the first mode, the simulation is run as a single program. The input deck structure for a Mode One run is shown in Figure C-1.

The second mode of execution allows an analyst to run the FORTRAN IV network reliability/availability routines and the SIMSCRIPT I.5 routines as two separate programs. Thus, an analyst can investigate the reliability, availability, and survivability aspects of a particular network topology without having to exercise the complete simulation. Conversely, an analyst can also investigate the other aspects of a network configuration's performance (Chapter V) without having to exercise the reliability/availability routines. In both cases considerable savings in memory space and CPU execution time requirements can be realized by running the simulation in Mode Two. The input deck structures for both types of Mode Two simulation runs are shown in Figure C-2 and Figure C-3 respectively.

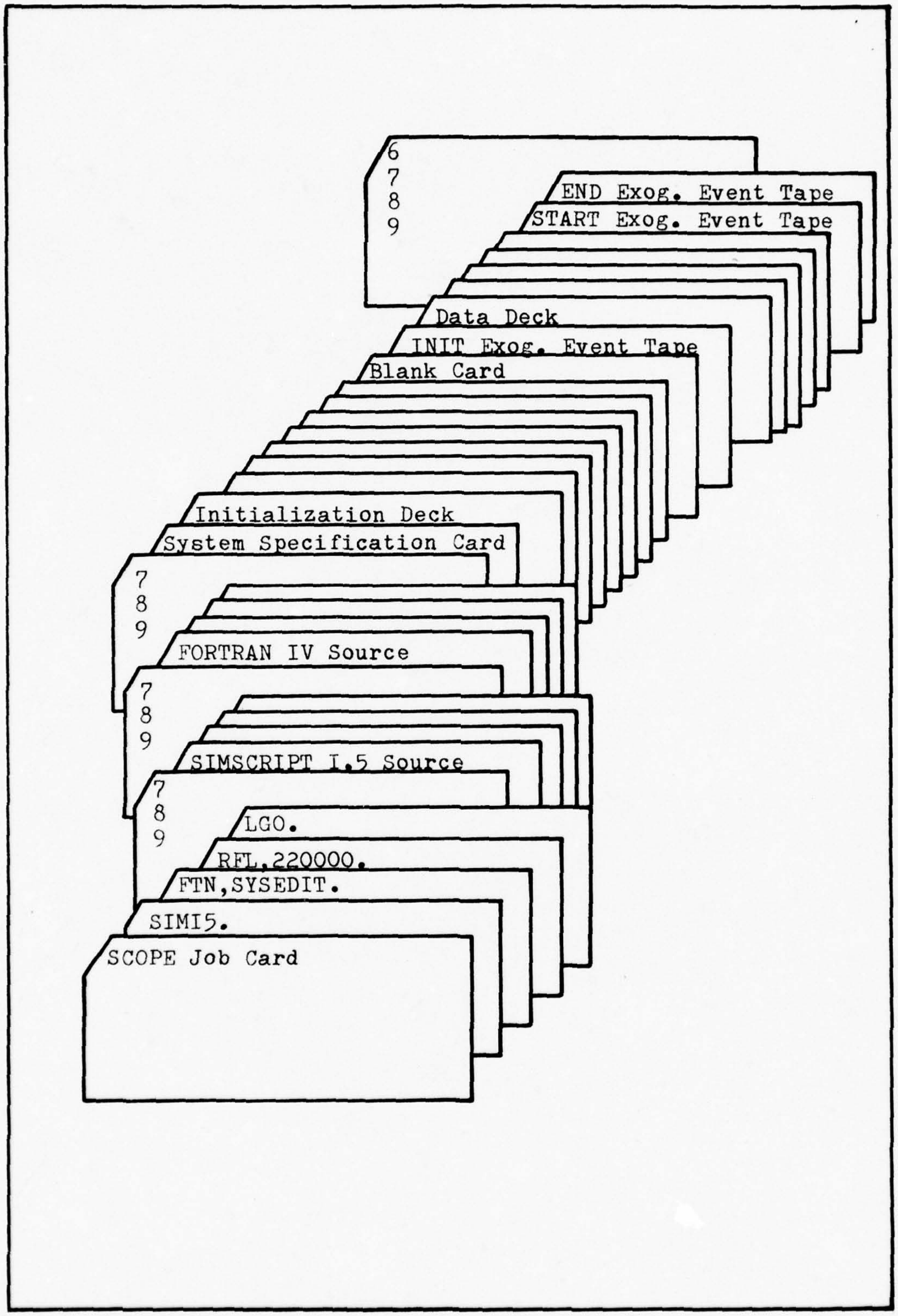


Fig C-1. Mode One Input Deck Structure

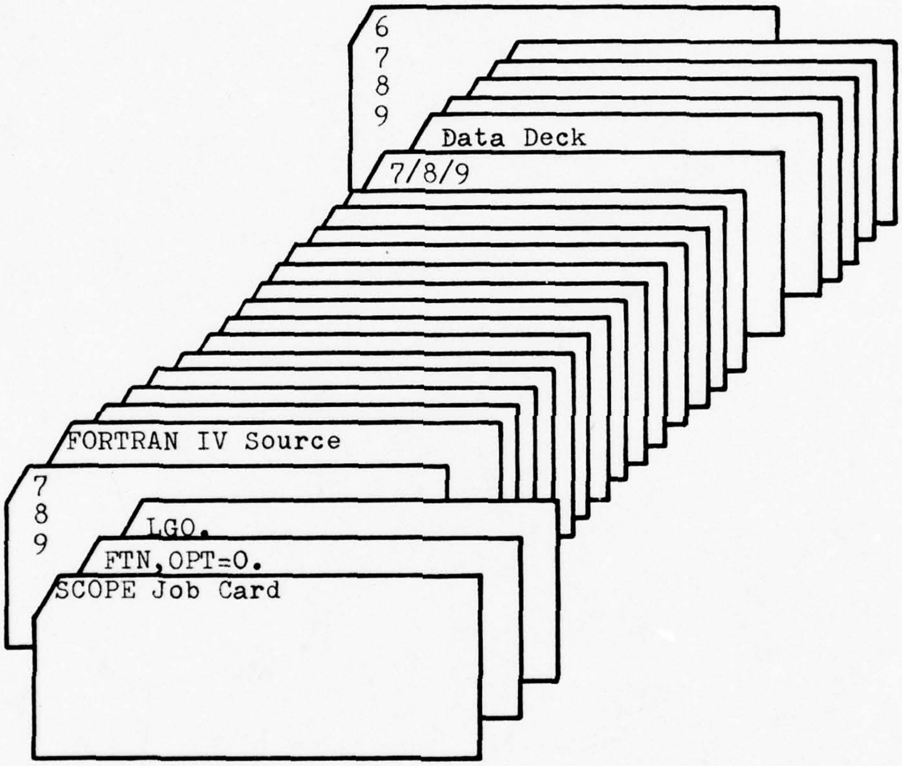


Fig C-2. Mode Two(R/A) Input Deck Structure

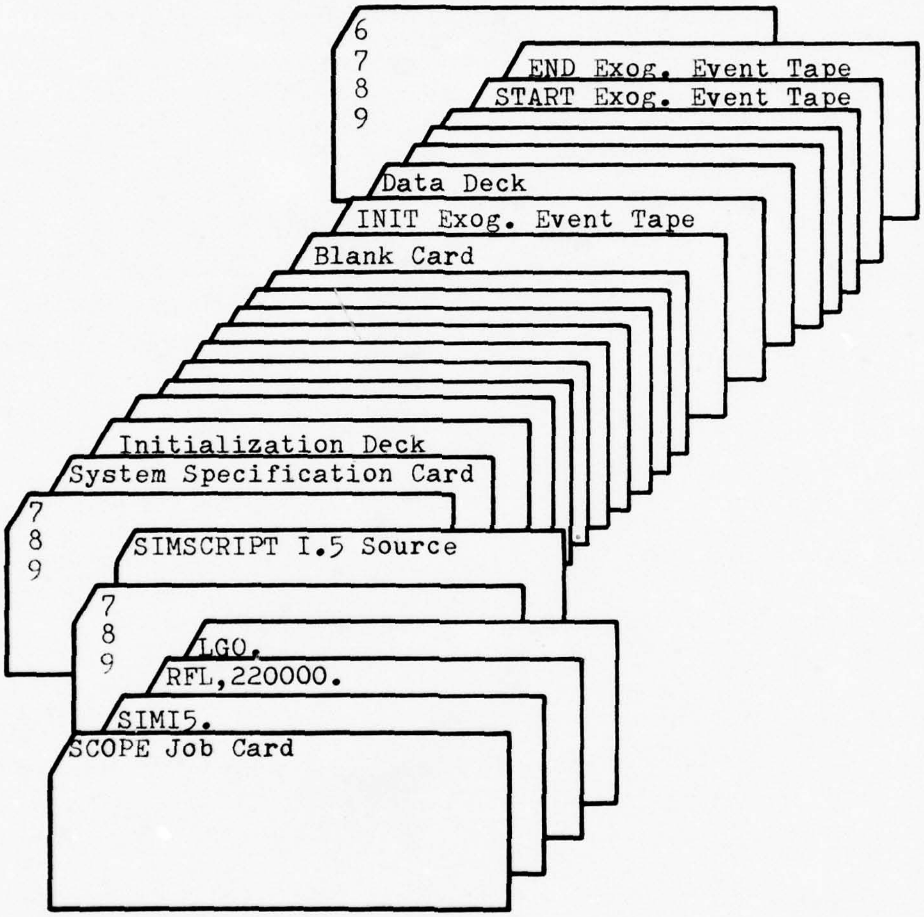


Fig C-3. Mode Two Input Deck Structure

Data Deck Structure and Specification. The data deck contains all the information required to drive the general computer network system simulation. The data deck is made up of the following eight data items:

1. INIT Exogenous Event Tape,
2. Simulation Specification Card,
3. Host Processor Specification Deck,
4. Topology Specification Deck,
5. Reliability/Availability Data Deck,
6. Transmission Link Specification Deck,
7. START Exogenous Event Tape, and
8. END Exogenous Event Tape.

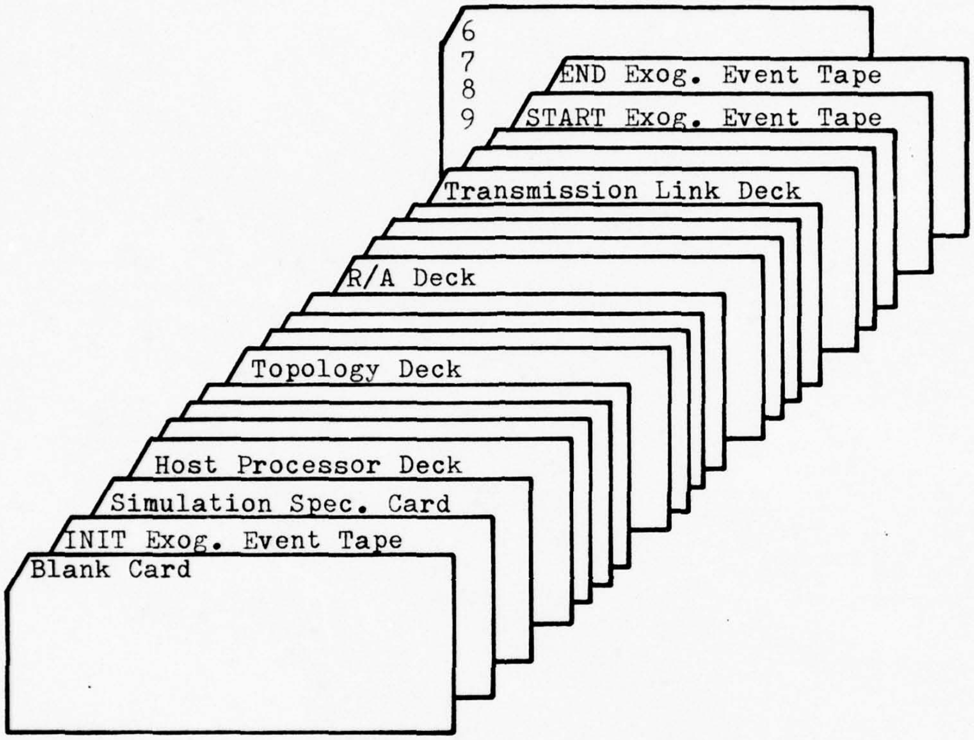
The structure of the data deck is shown in Figure C-4. The specification of each data item is as follows.

1. INIT Exogenous Event Tape.

Card Number	Column(s)	Format	Definition
1	1-3	I3	Identification number of Exogenous Event type; the number on the Events List. In this case "1".
"	4-7	I4	The "day" the Exogenous Event INIT is to occur.
"	8-10	I3	The"hour" of day the Exogenous Event INIT is to occur.
"	11-12	I2	The "minute" of the hour the Exogenous Event INIT is to occur.

2. Simulation Specification Card.

Card Number	Column(s)	Format	Definition
1	1-5	I5	The number of host processors in the simulated network.



NOTE: In a Mode Two simulation run the R/A Deck is removed and used seperately.

Fig C-4. Data Deck Structure

2. Simulation Specification Card (Cont).

Card Number	Column(s)	Format	Definition
1	5-10	I5	The number of host processors in the simulated network.
"	11-15	I5	The random number generator root.Presently set to one.
"	16-25	I10	The time in units at which the first periodic performance report will be made.
"	26-35	I10	The time in units between periodic performance reports.

3. Host Processor Specification Deck.

Card Number	Column(s)	Format	Definition
1	1-5	I5	The number of control points at a host processor.
"	6-10	I5	The number of I/O channels attached to a host processor.
"	11-15	I5	The total amount of main memory at a host processor.
"	16-20	I5	The maximum input queue length at a host processor.
"	21-25	I5	The word size in bits at a host processor.
2	1-5	I5	The number of job types input at a host processor.
3	1-5	I5	The cumulative probability of job type one occurring.
"	6-10	I5	BLANK
"	11-15	I5	The host processor location of the first nonorigin file request of the job type.
"	•	I5	" " " "
	•		
"	•		" " " "
"	40-45	I5	The host processor location of the ninth nonorigin file request of the job type.
"	45-50	I5	ZERO

A card like three is required for each job type characterized for a host processor. That is if there are N job types then there

3. Host Processor Specification Deck (Cont.).

must be N job type characterization cards.

Card Number	Column(s)	Format	Definition
2+N	1-10	D7.2	The mean job inter-arrival time in units at a host processor.
"	11-20	D7.2	The mean amount of main memory required by a job segment.
"	21-30	D7.2	The mean CPU execution time in units between I/O requests.
"	31-40	D7.2	The mean number of I/O requests per job segment.
"	41-50	D7.2	The mean time in units per I/O request.
"	51-60	D7.2	The per month hardware rental and maintenance cost for a host processor.
"	61-70	D7.2	The per month software rental and maintenance cost for a host processor.
3+N	1-10	A10	The nomenclature of a host processor.
"	11-20	A10	The geographic location of a host processor.
4+N	1-5	I5	The cumulative probability of a job having a priority of one.
"	•	"	" " " " "
"	• •	"	" " " " "
"	45-50	I5	The cumulative probability of a job having a priority of ten. If the probability is zero enter "0".
5+N	1-5	I5	The cumulative probability of a file request having only one job segment.

3. Host Processor Specifications Deck (Cont.).

Card Number	Column(s)	Format	Definition
"	• •	"	" " " " "
"	•	"	" " " " "
5+n	45	I5	The cumulative probability of a file request having ten job segments. If the probability is zero enter "0".

The preceding series of cards must be repeated for each host processor in the simulated computer network.

4. Topology Specification Deck.

Card Number	Column(s)	Format	Definition
1	1-5	I5	The identification numbers of the host processors on either end of the transmision link number.
"	6-10	I5	

A card like card one is required for each transmission link in the simulated computer network. That is, if there are L transmission links then there must be L transmission link location cards.

5. Reliability/Availability Data Deck.

Card Number	Column(s)	Format	Definition
1	1-5	I5	The number of transmission links in the simulated network.
"	6-10	I5	The identification number of the host processor from which the transmission paths under investigation originate.
"	11-15	I5	The identification number of the host processor to which the transmission paths under investigation terminate.
"	16-20	I5	The Network Reliability/Availability Report page number.

5. Reliability/Availability Data Deck (Cont.)

Card Number	Column(s)	Format	Definition
2	1-10	E10.4	The probability of successful communication for transmission link number one.
"	⋮	"	" " " "
"		"	" " " "
3	11-20	E10.4	The probability of successful communication link number ten.
"	21-30	E10.4	The epslion used in testing for reliability bound convergence.
4	1-10	E10.4	The communications availability probability of transmission link one.
"	⋮	"	" " " "
"		"	" " " "
5	11-20	E10.4	The communications availability probability of transmission link ten.
6	1-5	I5	Transmission link number one's identification number (1).
"	6-10	I5	The identification number of the first transmission to precede transmission link one.
"	⋮	"	" " " "
"		"	" " " "
"	45-50	I5	The identification number of the tenth transmission link to precede transmission link number one.

A card like card six is required for each transmission link in the simulated computer network. That is, if there are L transmission links then there must be L transmission link description cards. If a transmission link eminates from the origin host

5. Reliability/Availability Data Deck (Cont.)

processor then the absence of preceding transmission links is signified by a -1 in columns 6-10.

Card Number	Column(s)	Format	Definition
6+L	1-5	I5	The maximum number of transmission links allowed in the simulated computer network. Presently set to 25.
"	6-10	I5	The identification number of the first transmission link to terminate at the destination host processor.
"	•	"	" " " " " "
"	•	"	" " " " " "
"	45-50	I5	The identification number of the tenth transmission link to terminate at the destination host processor.

The preceding sequence of cards must be repeated for each host processor pair in the simulated computer network.

6. Transmission Link Specification Deck.

Card Number	Column(s)	Formats	Definition
1	1-5	D3.1	"00.0"
"	6-10	D3.1	The mean transmission time for the transmission link between host processor one and host processor two. If such a link does not exist set to "00.0".
"	•	"	" " " " "
"	•	"	" " " " "
"	45-50	D3.1	The mean transmission time for the transmission link between host processor one and host processor ten. If such a link does not exist set to "00.0".

6. Transmission Link Specification Deck (Cont.).

A card like card one is required for each host processor in the simulated computer network. That is, if there are N host processors then there must be N mean transmission time cards.

Card Number	Column(s)	Format	Definition
1+N	1-10	D7.2	The bandwidth in Kilobits/secs. of transmission link one.
"	11-20	D7.2	The cost per mile per month of transmission link one.
"	21-30	D7.2	The per month terminal service cost of transmission link one.
"	31-40	D7.2	The length in miles of transmission link one.

A card like card 1+N is required for each transmission link in the simulated computer network. That is,if there are L host processors then there must be L transmission link description cards.

7. START Exogenous Event Tape.

Card Number	Column(s)	Format	Definition
1	1-3	I3	Identification number of Exogenous Event type, in this case "2".
"	4-7	I4	The "day" the Exogenous Event START is to occur.
"	8-10	I3	The "hour" of day the Exogenous Event START is to occur.
"	11-12	I2	The "minute" of the hour the Exogenous Event START is to occur.

8. END Exogenous Event Tape.

Card Number	Column(s)	Format	Definition
1	1-3	I3	Identification number of Exogenous Event type, in this case "3".

8. END Exogenous Event Tape (Cont.).

Card Number	Column(s)	Format	Definition
1	4-7	I4	The "day" the Exogenous Event END is to occur.
"	8-10	I3	The "hour" of day the Exogenous Event END is to occur.
"	11-12	I2	The "minute" of the hour the Exogenous Event END is to occur.

Resource Requirements

The general computer network system simulation is written in CDC SIMSCRIPT I.5 Version 3 with the exception of Reliability/Availability routines which are written in CDC FORTRAN IV Extended.

The simulation as presently written requires 130,000 octal words of central memory to load and approximately 300,000 octal words of central memory to execute when run in Mode One. In Mode two the Reliability/Availability program requires 46,000 octal words of central memory to load and execute. The remainder of the simulation requires 61,000 octal words of central memory to load and approximately 220,000 octal words of central memory to execute. The difference between load and execution memory requirements, in both modes, is the dynamic memory allocation scheme employed by SIMSCRIPT I.5 in the creation and destruction of Temporary Entities. The exact amount of central memory that will be required to execute the simulation depends on the size and number of Temporary Entities that will be generated by the System. An upper limit on central memory to be used must be set through use of the RFL card. See Figure C-2. The central processor time required by both Modes is totally dependent on the characterization of the computer network and its environment and can vary over a wide range of values.

The amount of I/O time required by the general computer network system simulation is generally less than the Central Processor. The primary factor in determining how much I/O time might be required is the time interval between periodic performance reports. The smaller the time interval the greater the I/O required to generate the performance reports.

The only files used by the simulation are the default files input and output.

APPENDIX D

SIMULATION EXPERIMENTS

APPENDIX D

Simulation Experiments

This section contains the output reports for the five simulation experiments discussed in Chapter VII. These reports have been abbreviated to include only the characterization and final output reports for each experiment. This was done in order to conserve space. It should also be noted that the output reports presented are a typed copy of the original reports. This is due to the poor legibility of the originals.

OUTPUT REPORT

SIMULATION EXPERIMENT I

```
                     CHARACTERISTICS REPORT

                       PROCESSOR NUMBER 1               PAGE   1

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 360/40
PROCESSOR LOCATION - SITE A
PROCESSOR MAIN MEMORY - 256 KWORDS
NUMBER OF CONTROL POINTS - 17         NUMBER OF I/O CHANNELS  -  5
MAXIMUM INPUT QUEUE LENGTH - 34 JOBS PROCESSOR WORD SIZE - 32BITS

WORKLOAD CHARACTERISTICS:

JOB CUMULATIVE                    PROCESSOR FILES REQUESTED
TYPE PROBABILITY        1    2    3    4    5    6    7    8    9

 1       100            1    0    0    0    0    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME - 360.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS - 1.90 UNITS
MEAN TIME PER I/O REQUESTS - 2.10 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 22.50 KWORDS
MEAN NUMBER OF I/O REQUESTS PER JOB SEGMENT - 9.00

CUMULATIVE PROBABILITIES FOR JOBS/FILE REQUESTS:

                                         NUMBER
                     1    2    3    4    5    6    7    8    9   10
PRIORITY NUMBER    100    0    0    0    0    0    0    0    0    0
NO. OF SEGMENTS
PER FILE REQUEST   100    0    0    0    0    0    0    0    0    0
```

```
                         CHARACTERISTICS REPORT

                          PROCESSOR NUMBER 2                PAGE   2

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 370/165
PROCESSOR LOCATION - SITE B
PROCESSOR MAIN MEMORY - 512 KWORDS
NUMBER OF CONTROL POINTS - 21         NUMBER OF I/O CHANNELS  - 5
MAXIMUM INPUT QUEUE LENGTH - 42 JOBS PROCESSOR WORD SIZE - 32BITS

WORKLOAD CHARACTERISTICS:

JOB CUMULATIVE                        PROCESSOR FILES REQUIRED
TYPE PROBABILITY        1    2    3    4    5    6    7    8    9

 21       100           2    0    0    0    0    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME - 240.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS - 1.00 UNITS
MEAN TIME PER I/O REQUESTS - 1.50 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 37.00 KWORDS
MEAN NUMBER OF I/O REQUESTS PER JOB SEGMENT - 15.00

CUMULATIVE PROBABILITIES FOR JOBS/FILE REQUESTS:

                                       NUMBER
                     1    2    3    4    5    6    7    8    9    10

PRIORITY NUMBER     10   20   30   40   50   60   70   80   90   100

NO. OF SEGMENTS
PER FILE REQUEST100  0    0    0    0    0    0    0    0     0
```

```
                       CHARACTERISTICS REPORT

                         PROCESSOR NUMBER 3                PAGE   3

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 360/75
PROCESSOR LOCATION  - SITE C
PROCESSOR MAIN MEMORY - 256 KWORDS
NUMBER OF CONTROL POINTS - 15       NUMBER OF I/O CHANNELS -  4
MAXIMUM INPUT QUEUE LENGTH - 30 JOBS PROCESSOR WORD SIZE - 32BITS

WORKLOAD CHARACTERISTICS:

JOB CUMULATIVE                           PROCESSOR FILES REQUIRED
TYPE PROBABILITY        1    2    3    4    5    6    7    8    9
 41       100           3    0    0    0    0    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME - 480.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS - 1.60 UNITS
MEAN TIME PER I/O REQUESTS - 1.80 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 27.50 KWORDS
MEAN NUMBER OF I/O REQUESTS PER JOB SEGMENT - 8.00

CUMULATIVE PROBABILITIES FOR JOBS/FILE REQUESTS:

                                            NUMBER
                   1    2    3    4    5    6    7    8    9   10

PRIORITY NUMBER 100    0    0    0    0    0    0    0    0    0

NO. OF SEGMENTS
PER FILE REQUEST 10   20   30   40   50   60   70   80   90  100
```

```
                       CHARACTERISTICS REPORT

                        PROCESSOR NUMBER 4                PAGE   4

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 360/40
PROCESSOR LOCATION - SITE D
PROCESSOR MAIN MEMORY - 165 KWORDS
NUMBER OF CONTROL POINTS - 14         NUMBER OF I/O CHANNELS  -  3
MAXIMUM INPUT QUEUE LENGTH - 25 JOBS PROCESSOR WORD SIZE - 32BITS

WORKLOAD CHARACTERISTICS:

JOB CUMULATIVE                      PROCESSOR FILES REQUIRED
TYPE PROBABILITY         1    2    3    4    5    6    7    8    9
 61       100            4    0    0    0    0    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME - 340.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS - 2.10 UNITS
MEAN TIME PER I/O REQUESTS - 2.40 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 17.50 KWORDS
MEAN NUMBER OF I/OREQUESTS PER JOB SEGMENT - 5.00

CUMULATIVE PROBABILITIES FOR JOBS/FILE REQUESTS:

                                        NUMBER
                     1    2    3    4    5    6    7    8    9    10
PRIORITY NUMBER 100    0    0    0    0    0    0    0    0     0

NO. OF SEGMENTS
PER FILE REQUEST100    0    0    0    0    0    0    0    0     0
```

```
                              CHARACTERISTICS REPORT

                              TRANSMISSION FACILITIES              PAGE   5

TRANSMISSION LINK CHARACTERISTICS:

LINK     BANDWIDTH LENGTH   MILEAGE CHARGE   TERMINAL SERV. LINK COST
NUMBER     (KBS)   (MILES)  ($/MILE/MO.)        ($/MO.)     ($/MO.)

    1      56.00    17.65        4.54            500.00          580.13
    2      40.00    14.00        4.23            480.00          539.22
    3      24.00    26.25        4.07            420.00          526.84
    4      36.00    19.30        4.17            440.00          520.48

MEAN TRANSMISSION TIME PER LINK:

BETWEEN PROCESSOR N AND PROCESSOR M:

    N/M     1    2    3    4    5    6    7    8    9   10

     1    0.0 16.9  0.0 25.2  0.0  0.0  0.0  0.0  0.0  0.0
     2   16.9  0.0 22.9  0.0  0.0  0.0  0.0  0.0  0.0  0.0
     3    0.0 22.9  0.0 36.8  0.0  0.0  0.0  0.0  0.0  0.0
     4   25.2  0.0 36.8  0.0  0.0  0.0  0.0  0.0  0.0  0.0

NOTE: TIME IS IN UNITS.
```

```
              MONTHLY BASIC NETWORK CONFIGURATION COSTS REPORT

                                                              PAGE    6

PROCESSOR          HARDWARE                          SOFTWARE
  NUMBER    RENTAL AND MAINTENANCE           RENTAL AND MAINTENANCE
                  COST($/MO.)                         COST($/MO.)

    1            $   14209.00                         $   580.00
    2            $1940710.00                          $  1200.00
    3            $   70251.00                         $   714.00
    4            $   12439.00                         $   580.00
                 ------------                         ----------
            TOTAL$2037609.00                     TOTAL$  3074.00


  LINK            LINK COST
NUMBER            ($/MO.)
   1              $ 580.13
   2              $ 539.22
   3              $ 526.84
   4              $ 520.84
                 -----------
              TOTAL$2166.67

TOTAL MONTHLY COST OF BASIC NETWORK CONFIGURATION: $2042849.67
```

```
                         PROCESSOR OUTPUT REPORT

                          PROCESSOR NUMBER  1              PAGE    27

REPORT TIMES:

START REPORT TIME -       5 UNITS
END OF REPORT TIME - 21599 UNITS

PROCESSOR UTILIZATION STATISTICS:        THE NUMBER OF JOB SEGMENTS
                                         DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.0332           UNAVAILABLE C/PS -  0
CURRENT INPUT QUEUE LENGTH - 0 JOBS      UNAVAILABLE MEMORY -  0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH -  0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY           NUMBER OF             MEAN                      MEAN
 OF JOB          FILE REQUESTS        THROUGHPUT                    CPU
                   PROCESSED             TIME                      TIME

   1                  52              11978.96                    13.79

JOB TYPE STATISTICS:

JOB TYPE              MEAN               NUMBER
   ID           THROUGHPUT TIME            OF
                    PER JOB               JOBS

   1               11978.96                52

NOTE: TIME IS IN UNITS.
```

```
                 I/O CHANNEL UTILIZATION STATISTICS

                               REPORT

                        PROCESSOR NUMBER  1           PAGE     28

REPORT TIMES:

START OF REPORT TIME -      5 UNITS
END OF REPORT TIME -  21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                    PERCENT                 NUMBER OF
  NUMBER                     UTILIZATION              I/O REQUESTS

     1                          0.047                     467
     2                          0.003                      31
     3                          0.000                       0
     4                          0.000                       0
     5                          0.000                       0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS -  0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                         PROCESSOR OUTPUT REPORT

                           PROCESSOR NUMBER  2          PAGE     29

REPORT TIMES:

START OF REPORT TIME -       5 UNITS
END OF REPORT TIME -   21599 UNITS

PROCESSOR UTILIZATION STATISTICS:        THE NUMBER OF JOB SEGMENTS
                                         DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.0312           UNAVAILABLE C/PS -  0
CURRENT INPUT QUEUE LENGTH -  0 JOBS     UNAVAILABLE MEMORY -  0
MAXIMUM INPUT QUEUE LENGTH -  1 JOBS
MEAN INPUT QUEUE LENGTH -    0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY           NUMBER OF            MEAN                   MEAN
 OF JOB          FILE REQUESTS       THROUGHPUT                 CPU
                   PROCESSED            TIME                   TIME

    1                  7              14333.43                 3.14
    2                  8               8415.37                 8.00
    3                  9              10227.11                 6.78
    4                  8              12684.50                15.37
    5                  9               9111.11                 5.00
    6                  6               8312.67                 9.17
    7                  8              12115.62                13.75
    8                 14              10841.29                 5.00
    9                 13               9753.31                 8.00
   10                  4               4096.00                 4.75

JOB TYPE STATISTICS:

JOB TYPE             MEAN                NUMBER
   ID          THROUGHPUT TIME             OF
                   PER JOB                JOBS

    1             10289.92                 86

NOTE: TIME IS IN UNITS.
```

```
                    I/O CHANNEL UTILIZATION STATISTICS

                                  REPORT

                          PROCESSOR NUMBER  2             PAGE   30

REPORT TIMES:

START OF REPORT TIME -     5 UNITS
END OF REPRT TIME -  21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                  PERCENT                   NUMBER OF
   NUMBER                  UTILIZATION                I/O REQUESTS

     1                        0.081                      1154
     2                        0.004                        56
     3                        0.000                         0
     4                        0.000                         0
     5                        0.000                         0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS -  0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES -  0
```

```
                      PROCESSOR OUTPUT REPORT

                       PROCESSOR NUMBER 3              PAGE    31

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21599 UNITS

PROCESSOR UTILIZATION STATISTICS:       THE NUMBER OF JOB SEGMENTS
                                        DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.0798          UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS     UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY         NUMBER OF           MEAN                        MEAN
 OF JOB        FILE REQUESTS      THROUGHPUT                     CPU
                 PROCESSED           TIME                        TIME

    1                   39        11137.85                      44.21

JOB TYPE STATISTICS:

JOB TYPE             MEAN               NUMBER
  ID           THROUGHPUT TIME            OF
                   PER JOB               JOBS

   1              11137.89                39

NOTE: TIME IS IN UNITS.
```

```
                  I/O CHANNEL UTILIZATION STATISTICS

                                REPORT

                      PROCESSOR NUMBER  3            PAGE   32

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                PERCENT                  NUMBER OF
  NUMBER                 UTILIZATION              I/O REQUESTS

     1                      0.127                     1526
     2                      0.007                       82
     3                      0.000                        0
     4                      0.000                        0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                      PROCESSOR OUTPUT REPORT

                         PROCESSOR NUMBER 4             PAGE  33

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21599 UNITS

PROCESSOR UTILIZATION STATISTICS:       THE NUMBER OF JOB SEGMENTS
                                        DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.0151          UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS     UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY          NUMBER OF             MEAN                    MEAN
 OF JOB         FILE REQUESTS       THROUGHPUT                  CPU
                  PROCESSED             TIME                    TIME
   1                 39             11756.82                    8.36

JOB TYPE STATISTICS

JOB TYPE                MEAN                NUMBER
  ID            THROUGHPUT TIME               OF
                    PER JOB                  JOBS

   1             11756.82                     39

NOTE: TIME IS IN UNITS.
```

```
                    I/O CHANNEL UTILIZATION STATISTICS

                                 REPORT

                         PROCESSOR NUMBER  4            PAGE   34

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                   PERCENT                  NUMBER OF
  NUMBER                    UTILIZATION                I/O REQUESTS

     1                         0.021                      177
     2                         0.000                        0
     3                         0.000                        0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                TRANSMISSION LINK STATISTICS REPORT

                                                        PAGE   35

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

TUPLE REQUEST STATISTICS
FROM PROCESSOR N TO PROCESSOR M:

     N/M    1    2    3    4    5    6    7    8    9   10

      1     0    0    0    0    0    0    0    0    0     0
      2     0    0    0    0    0    0    0    0    0     0
      3     0    0    0    0    0    0    0    0    0     0
      4     0    0    0    0    0    0    0    0    0     0

TUPLE UTILIZATION STATISTICS
FROM PROCESSOR N TO PROCESSOR M:
     N/M    1    2    3    4    5    6    7    8    9   10

      1        0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
      2             0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
      3                  0.00 0.00 0.00 0.00 0.00 0.00 0.00
      4                       0.00 0.00 0.00 0.00 0.00 0.00
      5                            0.00 0.00 0.00 0.00 0.00
      6                                 0.00 0.00 0.00 0.00
      7                                      0.00 0.00 0.00
      8                                           0.00 0.00
      9                                                0.00

TRANSMISSION PROCESSING STATISTICS:

PROCESSOR  TRANSMISSION     MAXIMUM        MEAN         MEAN
  NUMBER     REQUESTS    OUTPUT QUEUE  OUTPUT QUEUE  TRANSMISSION
            ORIGINATED      LENGTH         LENGTH     DELAY TIME

      1          0            0            0.00          0.00
      2          0            0            0.00          0.00
      3          0            0            0.00          0.00
      4          0            0            0.00          0.00

NOTE: TIME IS IN UNITS.
```

```
+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
= THE COMPUTER NETWORK SIMULATION WAS TERMINATED +
+          SUCCESSFULLY AT TIME  6.0000.         =
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

OUTPUT REPORT

SIMULATION EXPERIMENT II

```
                      CHARACTERISTICS REPORT

                        PROCESSOR NUMBER  1              PAGE   1

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 360/40
PROCESSOR LOCATION - SITE A
PROCESSOR MAIN MEMORY - 256 KWORDS
NUMBER OF CONTROL POINTS - 17        NUMBER OF I/O CHANNELS   -  5
MAXIMUM INPUT QUEUE LENGTH - 34 JOBS PROCESSOR WORD SIZE - 32BITS

WORKLOAD CHARACTERISTICS:

JOB    CUMULATIVE                PROCESSOR FILES REQUIRED
TYPE   PROBABILITY      1    2    3    4    5    6    7    8    9
  1        100          1    2    3    4    1    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME - 360.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS - 1.90 UNITS
MEAN TIME PER I/O REQUESTS - 2.10 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 22.50 KWORDS
MEAN NUMBER OF I/O REQUESTS PER JOB SEGMENT - 9.00

CUMULATIVE PROBABILITIES FOR JOBS/FILE REQUESTS:

                                       NUMBER
                   1    2    3    4    5    6    7    8    9    10

PRIORITY NUMBER  100    0    0    0    0    0    0    0    0     0

NO. OF SEGMENTS
PER FILE REQUEST 100    0    0    0    0    0    0    0    0     0
```

```
                        CHARACTERISTICS REPORT

                          PROCESSOR NUMBER  2                PAGE   2

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 370/165
PROCESSOR LOCATION - SITE B
PROCESSOR MAIN MEMORY - 512 KWORDS
NUMBER OF CONTROL POINTS - 21         NUMBER OF I/O CHANNELS   -  5
MAXIMUM INPUT QUEUE LENGTH - 42 JOBS PROCESSOR WORD SIZE - 32BITS

WORKLOAD CHARACTERISTICS:

JOB   CUMULATIVE                   PROCESSOR FILES REQUIRED
TYPE  PROBABILITY         1    2    3    4    5    6    7    8    9

 21      100              2    0    0    0    0    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME - 240.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS - 1.00 UNITS
MEAN TIME PER I/O REQUEST - 1.50 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 37.00 KWORDS
MEAN NUMBER OF I/O REQUESTS PER JOB SEGMENT - 16.00

CUMULATIVE PROBABILITIES FOR JOBS/FILE REQUESTS:

                                        NUMBER
                    1    2    3    4    5    6    7    8    9   10
PRIORITY NUMBER   100    0    0    0    0    0    0    0    0    0

NO.OF SEGMENTS
PER FILE REQUEST 100     0    0    0    0    0    0    0    0    0
```

```
                    CHARACTERISTICS REPORT

                      PROCESSOR NUMBER  3             PAGE    3

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 360/75
PROCESSOR LOCATION - SITE C
PROCESSOR MAIN MEMORY - 256 KWORDS
NUMBER OF CONTROL POINTS - 15        NUMBER OF I/O CHANNELS  - 4
MAXIMUM INPUT QUEUE LENGTH - 30 JOBS PROCESSOR WORD SIZE - 32BITS

WORKLOAD CHARACTERISTICS:

JOB   CUMULATIVE                     PROCESSOR FILES REQUIRED
TYPE  PROBABILITY       1    2    3    4    5    6    7    8    9

 41       100           3    0    0    0    0    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME - 480.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS - 1.60 UNITS
MEAN TIME PER I/O REQUESTS - 1.80 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 27.50 KWORDS
MEAN NUMBER OF I/O REQUESTS PER JOB SEGMENT - 8.00

CUMULATIVE PROBABILITIES FOR JOBS/FILE REQUESTS:

                                            NUMBER
                     1    2    3    4    5    6    7    8    9   10

PRIORITY NUMBER    100    0    0    0    0    0    0    0    0    0

NO. OF SEGMENTS
PER FILE REQUEST   100    0    0    0    0    0    0    0    0    0
```

```
                         CHARACTERISTICS REPORT

                           PROCESSOR NUMBER  4             PAGE    4

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 360/40
PROCESSOR LOCATION - SITE D
PROCESSOR MAIN MEMORY - 165 KWORDS
NUMBER OF CONTROL POINTS - 14         NUMBER OF I/O CHANNELS  -  3
MAXIMUM INPUT QUEUE LENGTH - 28 JOBS PROCESSOR WORD SIZE - 32BITS

WORKLOAD CHARACTERISTICS:

JOB   CUMULATIVE                         PROCESSOR FILES REQUIRED
TYPE  PROBABILITY        1    2    3    4    5    6    7    8    9

 61       100            4    0    0    0    0    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME - 540.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS - 2.10 UNITS
MEAN TIME PER I/O REQUESTS - 2.40 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 17.50 KWORDS
MEAN NUMBER OF I/O REQUESTS PER JOB SEGMENT - 5.00

CUMULATIVE PROBABILITIES FOR JOBS FILE REQUESTS:

                                           NUMBER
                      1    2    3    4    5    6    7    8    9   10

PRIORITY NUMBER     100    0    0    0    0    0    0    0    0    0

NO. OF SEGMENTS
PER FILE REQUEST    100    0    0    0    0    0    0    0    0    0
```

```
                          CHARACTERISTICS REPORT

                          TRANSMISSION FACILITIES             PAGE   5

TRANSMISSION LINK CHARACTERISTICS:

LINK     BANDWIDTH LENGTH   MILEAGE CHARGE TERMINAL SERV. LINK COST
NUMBER     (KBS)   (MILES)  ($/MILE/MO.)      ($/MO.)      ($/MO.)

   1       56.00    17.65        4.54          500.00        580.13
   2       40.00    14.00        4.23          480.00        539.22
   3       24.00    26.25        4.07          420.00        526.84
   4       36.00    19.30        4.17          440.00        520.48

MEAN TRANSMISSION TIME PER LINK:

FROM PROCESSOR N TO PROCESSOR M:
       N/M    1     2     3     4     5     6     7     8     9    10

        1    0.0  14.9   0.0  23.2   0.0   0.0   0.0   0.0   0.0   0.0
        2   14.9   0.0  20.9   0.0  0.0    0.0   0.0   0.0   0.0   0.0
        3    0.0  20.9   0.0  34.8   0.0   0.0   0.0   0.0   0.0   0.0
        4   23.2   0.0  34.8   0.0   0.0   0.0   0.0   0.0   0.0   0.0

NOTE: TIME IS IN UNITS
```

```
          MONTHLY BASIC NETWORK CONFIGURATION COSTS REPORT

                                                        PAGE   6

PROCESSOR        HARDWARE                        SOFTWARE
 NUMBER    RENTAL AND MAINTENANCE         RENTAL AND MAINTENANCE
                COST ($/MO.)                     COST($/MO.)

    1          $   14209.00                  $ 580.00
    2          $1940710.00                   $1200.00
    3          $   70251.00                  $ 714.00
    4          $   12439.00                  $ 580.00
               ------------                  --------
         TOTAL $2037609.00             TOTAL $ 3074.00

LINK               LINK COST
NUMBER             ($/MO.)

   1               $ 580.13
   2               $ 539.22
   3               $ 526.84
   4               $ 520.48
                   --------
            TOTAL $2166.67

TOTAL MONTHLY COST OF BASIC NETWORK CONFIGURATION:  $2042849.67
```

```
                     PROCESSOR OUTPUT REPORT

                       PROCESSOR NUMBER   1            PAGE   27

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME  - 21599 UNITS

PROCESSOR UTILIZATION STATISTICS:      THE NUMBER OF JOB SEGMENTS
                                       DELAYED DUE TO:

PROCESSOR UTILIZATION - 0 JOBS         UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS    UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY        NUMBER OF             MEAN                      MEAN
 OF JOB       FILE REQUESTS        THROUGHPUT                    CPU
                PROCESSED             TIME                      TIME

    1             106               6004.85                    19.33

JOB STATISTICS:

JOB TYPE            MEAN               NUMBER
  ID          THROUGHPUT TIME            OF
                  PER JOB               JOBS

   1             11563.38                52

NOTE: TIME IS IN UNITS.
```

```
                    I/O CHANNEL UTILIZATION STATISTICS

                                  REPORT

                         PROCESSOR NUMBER  1              PAGE   28

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME  - 21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                  PERCENT                     NUMBER OF
  NUMBER                   UTILIZATION                  I/O REQUESTS

    1                         0.089                         948
    2                         0.005                          46
    3                         0.001                          10
    4                         0.000                           0
    5                         0.000                           0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                     PROCESSOR OUTPUT REPORT

                       PROCESSOR NUMBER   2            PAGE   29

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME   - 21599 UNITS

PROCESSOR UTILIZATION STATISTICS:    THE NUMBER OF JOB SEGMENTS
                                     DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.0601       UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS  UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY          NUMBER OF          MEAN                    MEAN
  OF JOB        FILE REQUESTS      THROUGHPUT                CPU
                  PROCESSED          TIME                    TIME

    1              143             6583.70                   9.07

JOB TYPE STATISTICS:

JOB TYPE                MEAN         NUMBER
   ID               THROUGHPUT TIME    OF
                       PER JOB        JOBS

    1               10477.71           91

NOTE: TIME IS IN UNITS.
```

```
                  I/O CHANNEL UTILIZATION STATISTICS

                               REPORT

                         PROCESSOR NUMBER 2               PAGE   30

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                PERCENT                  NUMBER OF
  NUMBER                 UTILIZATION               I/O REQUESTS

    1                       0.150                      2105
    2                       0.014                       213
    3                       0.000                         0
    4                       0.000                         0
    5                       0.000                         0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                    PROCESSOR OUTPUT REPORT

                       PROCESSOR NUMBER  3            PAGE    31

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21599 UNITS

PROCESSOR UTILIZATION STATISTICS:      THE NUMBER OF JOB SEGMENTS
                                       DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.0295         UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE.LENGTH - 0 JOBS    UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY          NUMBER OF          MEAN                     MEAN
 OF JOB         FILE REQUESTS     THROUGHPUT                  CPU
                  PROCESSED          TIME                     TIME

    1                89            3847.70                    7.15

JOB TYPE STATISTICS:

JOB TYPE              MEAN             NUMBER
   ID           THROUGHPUT TIME          OF
                    PER JOB             JOBS

   1             9192.08                 37

NOTE: TIME IS IN UNITS
```

```
                    I/O CHANNEL UTILIZATION STATISTICS

                                 REPORT

                          PROCESSOR NUMBER   3              PAGE    32

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                    PERCENT                    NUMBER OF
  NUMBER                     UTILIZATION                 I/O REQUESTS

    1                           0.045                         563
    2                           0.001                           7
    3                           0.000                           0
    4                           0.000                           0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                    PROCESSOR OUTPUT REPORT

                       PROCESSOR NUMBER  4               PAGE    33

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21599 UNITS

PROCESSOR UTILIZATION STATISTICS:      THE NUMBER OF JOB SEGMENTS
                                       DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.0339         UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS    UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY          NUMBER OF          MEAN                       MEAN
 OF JOB         FILE REQUESTS     THROUGHPUT                    CPU
                  PROCESSED          TIME                       TIME

    1                87           4459.10                       8.40

JOB TYPE STATISTICS:

JOB TYPE              MEAN            NUMBER
  ID            THROUGHPUT TIME         OF
                    PER JOB            JOBS

   1            10993.71                35

NOTE: TIME IS IN UNITS.
```

```
                    I/O CHANNEL UTILIZATION STATISTICS
                                 REPORT
                        PROCESSOR NUMBER   4             PAGE      34

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                     PERCENT                       NUMBER OF
 NUMBER                       UTILIZATION                   I/O REQUESTS

   1                             0.048                          433
   2                             0.000                            6
   3                             0.000                            0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                    TRANSMISSION LINK STATISTICS REPORT

                                                          PAGE   35

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

TUPLE REQUEST STATISTICS
FROM PROCESSOR N TO PROCESSOR M:

     N/M    1    2    3    4    5    6    7    8    9   10

      1     0   64    0   52    0    0    0    0    0    0
      2     0    0   60    0    0    0    0    0    0    0
      3     0    0    0   54    0    0    0    0    0    0
      4     0    0    0    0    0    0    0    0    0    0

TUPLE UTILIZATION STATISTICS:
FROM PROCESSOR N TO PROCESSOR M:

     N/M    1    2    3    4    5    6    7    8    9   10

      1       0.07 0.00 0.07 0.00 0.00 0.00 0.00 0.00 0.00
      2            0.08 0.00 0.00 0.00 0.00 0.00 0.00 0.00
      3                 0.10 0.00 0.00 0.00 0.00 0.00 0.00
      4                      0.00 0.00 0.00 0.00 0.00 0.00
      5                           0.00 0.00 0.00 0.00 0.00
      6                                0.00 0.00 0.00 0.00
      7                                     0.00 0.00 0.00
      8                                          0.00 0.00
      9                                               0.00

TRANSMISSION PROCESSING STATISTICS:

PROCESSOR TRANSMISSION    MAXIMUM        MEAN          MEAN
 NUMBER     REQUESTS   OUTPUT QUEUE  OUTPUT QUEUE  TRANSMISSION
           ORIGINATED     LENGTH        LENGTH      DELAY TIME

    1          54           1            1.00          0.78
    2          52           1            1.00          0.15
    3          52           1            1.00          5.48
    4          52           1            1.00          1.50

NOTE: TIME IS IN UNITS.
```

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
+ THE COMPUTER NETWORK SIMULATION WAS TERMINATED =
=          SUCCESSFULLY AT TIME  6.0000.         +
+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
```

OUTPUT REPORT

SIMULATION EXPERIMENT III

```
                          CHARACTERISTICS REPORT

                           PROCESSOR NUMBER  1              PAGE   1

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 360/40
PROCESSOR LOCATION - SITE A
PROCESSOR MAIN MEMORY - 256 KWORDS
NUMBER OF CONTROL POINTS - 17        NUMBER OF I/O CHANNELS  -  5
MAXIMUM INPUT QUEUE LENGTH - 34 JOBS PROCESSOR WORD SIZE - 32BITS

WORKLOAD CHARACTERISTICS:

JOB   CUMULATIVE                        PROCESSOR FILES REQUIRED
TYPE  PROBABILITY        1    2    3    4    5    6    7    8    9
  1        25            1    0    0    0    0    0    0    0    0
  2        50            1    2    1    0    0    0    0    0    0
  3        75            1    2    3    1    0    0    0    0    0
  4       100            1    2    3    4    1    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME - 360.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS - 1.90 UNITS
MEAN TIME PER I/O REQUESTS - 2.10 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 22.50 KWORDS
MEAN NUMBER OF I/O REQUESTS PER JOB SEGMENT - 9.00

CUMULATIVE PROBABILITIES FOR JOBS/FILE REQUESTS:

                                          NUMBER
                     1    2    3    4    5    6    7    8    9   10

PRIORITY NUMBER      0   10   30   80   90  100    0    0    0    0

NO. OF SEGMENTS
PER FILE REQUEST 100    0    0    0    0    0    0    0   0     0
```

```
                    CHARACTERISTICS REPORT

                      PROCESSOR NUMBER  2              PAGE    2

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 370/165
PROCESSOR LOCATION - SITE B
PROCESSOR MAIN MEMORY - 512 KWORDS
NUMBER OF CONTROL POINTS - 21         NUMBER OF I/O CHANNELS  - 5
MAXIMUM INPUT QUEUE LENGTH - 42 JOBS PROCESSOR WORD SIZE - 32BITS

WORKLOAD CHARACTERISTICS:

JOB   CUMULATIVE                     PROCESSOR FILES REQUIRED
TYPE  PROBABILITY        1    2    3    4    5    6    7    8    9

 21       25             2    0    0    0    0    0    0    0    0
 22       50             2    3    2    0    0    0    0    0    0
 23       75             2    3    4    2    0    0    0    0    0
 24      100             2    3    4    1    2    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME -240.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS - 1.00 UNITS
MEAN TIME PER I/O REQUESTS - 1.50 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 37.00 KWORDS
MEAN NUMBER OF I/O REQUESTS PER JOB SEGMENT - 15.00

CUMULATIVE PROBABILITIES FOR JOBS/FILE REQUESTS:

                                        NUMBER
                    1    2    3    4    5    6    7    8    9   10

PRIORITY NUMBER     0    0   70   90   95   100   0    0    0    0

NO. OF SEGMENTS
PER FILE REQUEST100      0    0    0    0     0   0    0    0    0
```

```
                         CHARACTERISTICS REPORT

                           PROCESSOR NUMBER  3              PAGE   3

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 360/75
PROCESSOR LOCATION - SITE C
PROCESSOR MAIN MEMORY - 256 KWORDS
NUMBER OF CONTROL POINTS - 15       NUMBER OF I/O CHANNELS - 4
MAXIMUM INPUT QUEUE LENGTH - 30 JOBS PROCESSOR WORD SIZE 32BITS

WORKLOAD CHARACTERISTICS:

JOB   CUMULATIVE                        PROCESSOR FILES REQUIRED
TYPE  PROBABILITY         1    2    3    4    5    6    7    8    9

 41        25             3    0    0    0    0    0    0    0    0
 42        50             3    4    3    0    0    0    0    0    0
 43        75             3    4    1    3    0    0    0    0    0
 44       100             3    4    1    2    3    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME - 480.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS- 1.60 UNITS
MEAN TIME PER I/O REQUESTS - 1.80 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 27.50 KWORDS
MEAN NUMBER OF I/O REQUESTS PER JOB SEGMENT - 8.00

CUMULATIVE PROBABILITIES FOR JOBS/FILE REQUESTS:

                                          NUMBER
                     1    2    3    4    5    6    7    8    9   10

PRIORITY NUMBER     20   60   90   95  100    0    0    0    0    0

NO. OF SEGMENTS
PER FILE REQUEST   100    0    0    0    0    0   0     0    0    0
```

```
                        CHARACTERISTICS REPORT

                          PROCESSOR NUMBER  4               PAGE    4

PROCESSOR CHARACTERISTICS:

PROCESSOR TYPE - 360/40
PROCESSOR LOCATION - SITE D
PROCESSOR MAIN MEMORY - 165 KWORDS
NUMBER OF CONTROL POINTS - 14        NUMBER OF I/O CHANNELS  - 3
MAXIMUM INPUT QUEUE LENGTH - 28 JOBS PROCESSOR WORD SIZE - 32BITS

WORKLOAD CHARACTERISTICS:

JOB   CUMULATIVE                     PROCESSOR FILES REQUIRED
TYPE  PROBABILITY         1    2    3    4    5    6    7    8    9

 61       25              4    0    0    0    0    0    0    0    0
 62       50              4    1    4    0    0    0    0    0    0
 63       75              4    1    2    4    0    0    0    0    0
 64      100              4    1    2    3    4    0    0    0    0

MEAN JOB INTER-ARRIVAL TIME -540.00 UNITS
MEAN CPU TIME BETWEEN I/O REQUESTS - 2.10 UNITS
MEAN TIME PER I/O REQUESTS - 2.40 UNITS
MEAN MEMORY REQUIRED PER JOB SEGMENT - 17.50 KWORDS
MEAN NUMBER OF I/O REQUESTS PER JOB SEGMENT - 5.00

CUMULATIVE PROBABILITIES FOR JOBS/FILE REQUESTS:

                                          NUMBER

                      1    2    3    4    5    6    7    8    9   10

PRIORITY NUMBER       0    0    0   85   90   95  100    0    0    0

NO. OF SEGMENTS
PER FILE REQUEST    100    0    0    0    0    0    0    0    0    0
```

```
                         CHARACTERISTICS REPORT

                         TRANSMISSION FACILITIES              PAGE   5

TRANSMISSION LINK CHARACTERISTICS:

LINK    BANDWIDTH LENGTH  MILEAGE CHARGE TERMINAL SERV.  LINK COST
NUMBER    (KBS)   (MILES)  ($/MILE/MO.)    ($/MO.)        ($/M).)
   1      56.00    17.65         4.54         500.00        580.13
   2      40.00    14.00         4.23         480.00        539.22
   3      24.00    26.25         4.07         420.00        526.84
   4      36.00    19.30         4.71         440.00        520.48

MEAN TRANSMISSION TIME PER LINK:
FROM PROCESSOR N TO PROCESSOR M:
      N/M    1     2     3     4     5     6     7     8     9    10

       1    0.0  16.9   0.0  25.2   0.0   0.0   0.0   0.0   0.0   0.0
       2   16.9   0.0  22.9   0.0   0.0   0.0   0.0   0.0   0.0   0.0
       3    0.0  22.9   0.0  36.8   0.0   0.0   0.0   0.0   0.0   0.0
       4   25.2   0.0  36.8   0.0   0.0   0.0   0.0   0.0   0.0   0.0

NOTE: TIME IS IN UNITS.
```

```
               MONTHLY BASIC NETWORK CONFIGURATION COSTS REPORT

                                                              PAGE   6

PROCESSOR          HARDWARE                         SOFTWARE
 NUMBER    RENTAL AND MAINTENANCE            RENTAL AND MAINTENANCE
                COSTS($/MO.)                       COST($/MO.)

    1          $   14209.00                         $   580.00
    2          $1940710.00                          $  1200.00
    3          $   70251.00                         $   714.00
    4          $   12439.00                         $   580.00
               ------------                         ----------
          TOTAL$2037609.00                     TOTAL$  3074.00

LINK               LINK COST
NUMBER              ($/MO.)

 1                 $ 580.13
 2                 $ 539.22
 3                 $ 526.84
 4                 $ 520.48
                   --------
              TOTAL$2166.67

TOTAL MONTHLY COST OF BASIC NETWORK CONFIGURATION: $2042849.67
```

```
                      PROCESSOR OUTPUT REPORT

                         PROCESSOR NUMBER  1           PAGE    27

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21598 UNITS

PROCESSOR UTILIZATION STATISTICS:     THE NUMBER OF JOB SEGMENTS
                                      DELAYED DUE TO:

PROCESSOR UTILIZATION - 1254          UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS   UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 2 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY          NUMBER OF           MEAN                 MEAN
 OF JOB         FILE REQUESTS      THROUGHPUT               CPU
                  PROCESSED           TIME                 TIME

    1                 9               57.22               26.22
    2                15             2978.60               19.53
    3                35             2700.91               16.31
    4                72             4991.56               19.07
    5                 7             4760.29                8.71
    6                12             4685.92                9.58

JOB TYPE STATISTICS:

JOB TYPE             MEAN          NUMBER
  ID           THROUGHPUT TIME       OF
                  PER JOB           JOBS

     1           13882.07            14
     2           12398.08            13
     3            7543.75            12
     4           11839.75            12

NOTE: TIME IS IN UNITS.
```

```
                 I/O CHANNEL UTILIZATION STATISTICS

                              REPORT

                       PROCESSOR NUMBER  1               PAGE   28

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21598 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                  PERCENT                    NUMBER OF
  NUMBER                   UTILIZATION                 I/O REQUESTS

     1                        0.122                        1240
     2                        0.014                         163
     3                        0.001                          11
     4                        0.000                           3
     5                        0.000                           0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                      PROCESSOR OUTPUT REPORT

                         PROCESSOR NUMBER  2             PAGE    29

REPORT TIMES

START REPORT TIME - 5 UNITS
END REPORT TIME - 21598 UNITS

PROCESSOR UTILIZATION STATISTICS:       THE NUMBER OF JOB SEGMENTS
                                        DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.1390          UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS     UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 2 JOBS
MEAN INPUT QUEUE LENGTH - 1.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY           NUMBER OF           MEAN                 MEAN
 OF JOB          FILE REQUESTS      THROUGHPUT              CPU
                   PROCESSED           TIME                 TIME

    1                  7             2434.29               16.57
    2                 10               80.10               10.80
    3                111             5577.88               12.86
    4                 85             4776.80               13.45
    5                  9             5886.78                5.78
    6                 14             2012.14               10.57

JOB TYPE STATISTICS:

JOB TYPE              MEAN             NUMBER
  ID            THROUGHPUT TIME          OF
                   PER JOB              JOBS

     1            13092.46               28
     2            11296.50               26
     3            10983.91               23
     4             8571.87               23

NOTE: TIME IS IN UNITS.
```

```
                I/O CHANNEL UTILIZATION STATISTICS

                              REPORT

                    PROCESSOR NUMBER  2              PAGE   30

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21598 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                PERCENT                 NUMBER OF
  NUMBER                 UTILIZATION              I/O REQUESTS

    1                       0.229                      3291
    2                       0.038                       530
    3                       0.008                       112
    4                       0.002                        23
    5                       0.000                         1

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                        PROCESSOR OUTPUT REPORT

                           PROCESSOR NUMBER  3            PAGE    31

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21598 UNITS

PROCESSOR UTILIZATION STATISTICS:     THE NUMBER OF JOB SEGMENTS
                                      DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.0831        UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS   UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY          NUMBER OF         MEAN                       MEAN
 OF JOB         FILE REQUESTS    THROUGHPUT                    CPU
                  PROCESSED         TIME                       TIME

  1                 20            5659.20                     13.10
  2                 34            6343.35                     12.35
  3                 61            1651.74                     11.72
  4                 45             135.40                      7.93
  5                  4              21.75                      7.50

JOB TYPE STATISTICS:

JOB TYPE            MEAN            NUMBER
  ID          THROUGHPUT TIME        OF
                  PER JOB           JOBS

  1               9999.00             9
  2              12628.45            11
  3              12504.75             4
  4              11940.15            13

NOTE: TIME IS IN UNITS.
```

```
                 I/O CHANNEL UTILIZATION STATISTICS

                               REPORT

                       PROCESSOR NUMBER  3              PAGE   32

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21598 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                    PERCENT                   NUMBER OF
   NUMBER                    UTILIZATION                I/O REQUESTS

     1                          0.101                       1181
     2                          0.011                        149
     3                          0.001                          9
     4                          0.000                          0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS-0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                    PROCESSOR OUTPUT REPORT

                       PROCESSOR NUMBER  4            PAGE   33

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21598 UNITS

PROCESSOR UTILIZATION STATISTICS:    THE NUMBER OF JOB SEGMENTS
                                     DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.0488       UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS  UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH- 1 JOBS
MEAN INPUT QUEUE LENGTH - 1.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY          NUMBER OF          MEAN                      MEAN
 OF JOB         FILE REQUESTS     THROUGHPUT                   CPU
                  PROCESSED          TIME                      TIME

   1                  7              43.43                     5.43
   2                 16              98.87                     6.19
   3                 31             106.26                     6.00
   4                 79            5519.08                     8.27
   5                  5            2947.80                    10.40
   6                  5             104.80                     2.80
   7                  4            8603.75                     2.75

JOB TYPE STATISTICS:

JOB TYPE           MEAN              NUMBER
  ID         THROUGHPUT TIME           OF
                 PER JOB              JOBS

   1             12400.69              13
   2             12336.89               9
   3             15095.67               6
   4             13701.78               9

NOTE: TIME IS IN UNITS.
```

```
                    I/O CHANNEL UTILIZATION STATISTICS

                                  REPORT

                         PROCESSOR NUMBER  4                PAGE  34

REPORT TIMES

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21598 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                  PERCENT                      NUMBER OF
   NUMBER                  UTILIZATION                   I/O REQUESTS

     1                        0.063                        561
     2                        0.006                         55
     3                        0.001                          6

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

AD-A039 772 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 9/2
A GENERAL COMPUTER NETWORK SIMULATION MODEL.(U)
MAR 77 H M WARREN
UNCLASSIFIED AFIT/GCS/EE/77-3 NL

4 OF 4
ADA039772

END
DATE
FILMED
6-77

```
                 TRANSMISSION LINK STATISTICS REPORT

                                                      PAGE 35

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21598 UNITS

TUPLE REQUEST STATISTICS
FROM PROCESSOR N AND PROCESSOR M:

     N/M    1    2    3    4    5    6    7    8    9   10

      1     0  173    0  141    0    0    0    0    0    0
      2     0    0  186    0    0    0    0    0    0    0
      3     0    0    0  118    0    0    0    0    0    0
      4     0    0    0    0    0    0    0    0    0    0

TUPLE UTILIZATION STATISTICS
FROM PROCESSOR N AND PROCESSOR M:

     N/M    1    2    3    4    5    6    7    8    9   10

      1       0.27 0.00 0.30 0.00 0.00 0.00 0.00 0.00 0.00
      2            0.26 0.00 0.00 0.00 0.00 0.00 0.00 0.00
      3                 0.28 0.00 0.00 0.00 0.00 0.00 0.00
      4                      0.00 0.00 0.00 0.00 0.00 0.00
      5                           0.00 0.00 0.00 0.00 0.00
      6                                0.00 0.00 0.00 0.00
      7                                     0.00 0.00 0.00
      8                                          0.00 0.00
      9                                               0.00

TRANSMISSION PROCESSING STATISTICS:
PROCESSOR  TRANSMISSION    MAXIMUM        MEAN          MEAN
   NUMBER    REQUESTS    OUTPUT QUEUE  OUTPUT QUEUE  TRANSMISSION
            ORIGINATED      LENGTH        LENGTH      DELAY TIME

    1          101            3            1.08          14.64
    2          138            2            1.03           8.56
    3          134            4            1.10          17.99
    4          110            4            1.19          20.78

NOTE: TIME IS IN UNITS.
```

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+_+=+
+ THE COMPUTER NETWORK SIMULATION WAS TERMINATED +
=           SUCCESSFULLY AT TIME  6.0000.        =
+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+_+=+=+=
```

NETWORK RELIABILITY/AVAILABILITY REPORT

SIMULATION EXPERIMENT III

```
                    NETWORK RELIABILITY/AVAILABILITY REPORT

                         NETWORK AS A WHOLE              PAGE    7

PESSIMISTIC NETWORK RELIABILITY IS  .98957E+00

MEAN NETWORK RELIABILITY IS  .99172E+00

PESSIMISTIC NETWORK AVAILABILITY IS  .83430E+00

MEAN NETWORK AVAILABILITY IS  .83689E+00
```

OUTPUT REPORT

SIMULATION EXPERIMENT IV

```
                          CHARACTERISTICS REPORT

                          TRANSMISSION FACILITIES           PAGE    5

TRANSMISSION LINK CHARACTERISTICS:

LINK    BANDWIDTH LENGTH   MILEAGE CHARGE TERMINAL SERV.   LINK COST
NUMBER    (KBS)   (MILES)  ($/MILE MO.)        ($/MO.)      ($/MO.)

  1      56.00     17.65     4.54             500.00        580.13
  2      40.00     14.00     4.23             480.00        539.22
  3      56.00     29.80     4.54             500.00        635.29

MEAN TRANSMISSION TIME PER LINK:

FROM PROCESSOR N AND PROCESSOR M:

      N/M    1    2    3    4    5    6    7    8    9    10

       1    0.0 16.9  0.0  0.0  0.0  0.0  0.0  0.0  0.0   0.0
       2   16.9  0.0 22.9 16.9  0.0  0.0  0.0  0.0  0.0   0.0
       3    0.0 22.9  0.0  0.0  0.0  0.0  0.0  0.0  0.0   0.0
       4    0.0 16.9  0.0  0.0  0.0  0.0  0.0  0.0  0.0   0.0

NOTE: TIME IS IN UNITS.
```

```
          MONTHLY BASIC NETWORK CONFIGURATION COSTS REPORT

                                                        PAGE   6

PROCESSOR         HARDWARE                          SOFTWARE
 NUMBER     RENTAL AND MAINTENANCE            RENTAL AND MAINTENANCE
                 COST($/MO.)                         COST($/MO.)

    1           $  14209.00                       $  580.00
    2           $1940710.00                       $ 1200.00
    3           $  70251.00                       $  714.00
    4           $  12439.00                       $  580.00
                -----------                       ---------
           TOTAL$2037609.00                  TOTAL$ 3074.00

LINK            LINK COST
NUMBER           ($/MO.)

    1           $  580.13
    2           $  539.22
    3           $  635.29
                ---------
           TOTAL$ 1754.64

TOTAL MONTHLY COST OF BASIC NETWORK CONFIGURATION:  $2042437.64
```

```
                        PROCESSOR OUTPUT REPORT

                          PROCESSOR NUMBER  1          PAGE    27

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21598 UNITS

PROCESSOR UTILIZATION STATISTICS:      THE NUMBER OF JOB SEGMENTS
                                       DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.1190         UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS    UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY         NUMBER OF          MEAN                    MEAN
 OF JOB        FILE REQUESTS     THROUGHPUT                 CPU
                 PROCESSED          TIME                    TIME

    1                 8            122.00                  37.12
    2                22           4058.45                  10.73
    3                41           2427.39                  12.59
    4                69           3703.12                  17.04
    5                14           4320.21                  17.07
    6                 9           6557.56                  10.67

JOB TYPE STATISTICS:

JOB TYPE            MEAN           NUMBER
   ID         THROUGHPUT TIME        OF
                 PER JOB            JOBS

    1           11884.08             13
    2            9124.90             10
    3           11411.33              9
    4           10668.50             18

NOTE: TIME IS IN UNITS.
```

```
               I/O CHANNEL UTILIZATION STATISTICS

                             REPORT

                       PROCESSOR NUMBER  1                PAGE   28

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21598 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                    PERCENT                 NUMBER OF
   NUMBER                    UTILIZATION              I/O REQUESTS

     1                          0.119                    1285
     2                          0.011                     114
     3                          0.001                       6
     4                          0.000                       0
     5                          0.000                       0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES-0
```

```
                    PROCESSOR OUTPUT REPORT

                       PROCESSOR NUMBER  2            PAGE   29

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21598 UNITS

PROCESSOR UTILIZATION STATISTICS:      THE NUMBER OF JOB SEGMENTS
                                       DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.1198         UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS    UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 1.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY         NUMBER OF         MEAN                    MEAN
 OF JOB        FILE REQUESTS     THROUGHPUT                CPU
                 PROCESSED         TIME                    TIME

    1                8            2999.37                 13.00
    2               11              85.73                 10.45
    3              113            5546.33                 11.66
    4               66            2678.56                 12.03
    5               12            2669.67                 14.58
    6                3            2360.75                 10.12

JOB TYPE STATISTICS:

JOB TYPE           MEAN             NUMBER
  ID         THROUGHPUT TIME          OF
                 PER JOB             JOBS

    1            10948.40              15
    2            10537.32              25
    3            11827.30              23
    4             9791.58              19

NOTE: TIME IS IN UNITS.
```

```
                    I/O CHANNEL UTILIZATION STATISTICS

                                  REPORT

                         PROCESSOR NUMBER  2               PAGE    30

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21598 UNITS

I/O CHANNEL UTILIZATION STATISTICS:
I/O CHANNEL                     PERCENT                  NUMBER OF
NUMBER                        UTILIZATION               I/O REQUEST
    1                            0.200                     2879
    2                            0.032                      487
    3                            0.007                      111
    4                            0.001                       11
    5                            0.000                        3

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                    PROCESSOR OUTPUT REPORT

                      PROCESSOR NUMBER  3           PAGE    31

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21598 UNITS

PROCESSOR UTILIZATION STATISTICS:      THE NUMBER OF JOB SEGMENTS
                                       DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.1091         UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS    UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 2 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY         NUMBER OF          MEAN                       MEAN
 OF JOB        FILE REQUESTS     THROUGHPUT                    CPU
                 PROCESSED          TIME                        TIME

   1                31             5036.61                     19.03
   2                42             5430.71                     12.48
   3                73             1930.15                     10.53
   4                34              234.29                      8.38
   5                13             2338.15                     13.23

JOB TYPE STATISTICS:

JOB TYPE             MEAN           NUMBER
  ID           THROUGHPUT TIME        OF
                   PER JOB           JOBS

   1               6216.71            17
   2              11827.53            15
   3              10421.82            11
   4              12898.77            13

NOTE: TIME IS IN UNITS.
```

```
                 I/O CHANNEL UTILIZATION STATISTICS

                               REPORT

                      PROCESSOR NUMBER  3            PAGE   32

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21598  UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                  PERCENT                 NUMBER OF
   NUMBER                  UTILIZATION              I/O REQUESTS

     1                        0.133                  1565
     2                        0.012                   155
     3                        0.000                     4
     4                        0.000                     0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                      PROCESSOR OUTPUT REPORT

                         PROCESSOR NUMBER  4            PAGE   33

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21598 UNITS

PROCESSOR UTILIZATION STATISTICS:       THE NUMBER OF JOB SEGMENTS
                                        DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.0688          UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS     UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 2 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY           NUMBER OF          MEAN                 MEAN
 OF JOB          FILE REQUESTS      THROUGHPUT             CPU
                   PROCESSED           TIME                TIME

   1                  15              106.27               5.07
   2                  17              105.06               5.06
   3                  41              102.66               8.51
   4                  79             5377.13              10.22
   5                  12             3077.25               9.58
   6                   5             2852.20               5.00
   7                   3            12851.00               9.33

JOB TYPE STATISTICS:

JOB TYPE             MEAN              NUMBER
   ID          THROUGHPUT TIME           OF
                   PER JOB              JOBS

   1              14024.43                7
   2              12821.47               15
   3              11607.90               10
   4              13707.12                8

NOTE: TIME IS IN UNITS.
```

```
                I/O CHANNEL UTILIZATION STATISTICS

                              REPORT

                       PROCESSOR NUMBER  4             PAGE   34

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21598 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                  PERCENT                NUMBER OF
  NUMBER                   UTILIZATION             I/O REQUESTS
     1                        0.075                   708
     2                        0.006                    37
     3                        0.000                     2

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                  TRANSMISSION LINK STATISTICS REPORT

                                                        PAGE   35

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21598 UNITS

TUPLE REQUEST STATISTICS
FROM PROCESSOR N AND PROCESSOR M:

     N/M    1    2    3    4    5    6    7    8    9    10

      1     0  228    0    0    0    0    0    0    0     0
      2     0    0  284  264    0    0    0    0    0     0
      3     0    0    0    0    0    0    0    0    0     0

TUPLE UTILIZATION STATISTICS
FROM PROCESSOR N AND PROCESSOR M:

     N/M    1    2    3    4    5    6    7    8    9    10
      1       0.29 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
      2            0.39 0.41 0.00 0.00 0.00 0.00 0.00 0.00
      3                 0.00 0.00 0.00 0.00 0.00 0.00 0.00
      4                      0.00 0.00 0.00 0.00 0.00 0.00
      5                           0.00 0.00 0.00 0.00 0.00
      6                                0.00 0.00 0.00 0.00
      7                                     0.00 0.00 0.00
      8                                          0.00 0.00
      9                                               0.00

TRANSMISSION PROCESSING STATISTICS:

PROCESSOR  TRANSMISSION    MAXIMUM        MEAN          MEAN
 NUMBER      REQUESTS    OUTPUT QUEUE  OUTPUT QUEUE  TRANSMISSION
            ORIGINATED      LENGTH        LENGTH      DELAY TIME

    1          114            2          1.14          20.68
    2          136            3          1.07          17.43
    3          142            6          1.35          46.97
    4          132            5          1.47          58.20

NOTE: TIME IS IN UNITS.
```

```
+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
= THE COMPUTER NETWORK SIMULATION WAS TERMINATED +
+          SUCCESSFULLY AT TIME  6.0000,         =
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

NETWORK RELIABILITY/AVAILABILITY REPORT

SIMULATION EXPERIMENT IV

```
              NETWORK RELIABILITY/AVAILABILITY REPORT
                        NETWORK AS A WHOLE             PAGE   7

PESSIMISTIC NETWORK RELIABILITY IS  .86480E+00

MEAN NETWORK RELIABILITY IS  .91173E+00

PESSIMISTIC NETWORK AVAILABILITY IS  .83009E+00

MEAN NETWORK AVAILABILITY IS  .88486E+00
```

OUTPUT REPORT

SIMULATION EXPERIMENT V

```
                              CHARACTERISTICS REPORT

                              TRANSMISSION FACILITIES          PAGE     5

TRANSMISSION LINK CHARACTERISTICS:

LINK   BANDWIDTH LENGTH  MILEAGE CHARGE TERMINAL SERV.  LINK COST
NUMBER   (KBS)   (MILES)  ($/MILE MO.)   ($/MO.)          ($/MO.)

  1      56.00    17.65      4.54          500.00          580.13
  2      40.00    14.00      4.23          480.00          539.22
  3      24.00    26.25      4.07          420.00          526.84
  4      36.00    19.30      4.17          440.00          520.48
  5      56.00    29.80      4.54          500.00          635.29

MEAN TRANSMISSION TIME PER LINK:

FROM PROCESSOR N TO PROCESSOR M:
     N/M    1    2    3    4    5    6    7    8    9    10

      1    0.0 16.9  0.0 25.2  0.0  0.0  0.0  0.0  0.0   0.0
      2   16.9  0.0 22.9 16.9  0.0  0.0  0.0  0.0  0.0   0.0
      3    0.0 22.9  0.0 36.8  0.0  0.0  0.0  0.0  0.0   0.0
      4   25.2 16.9 36.8  0.0  0.0  0.0  0.0  0.0  0.0   0.0

NOTE: TIME IS IN UNITS:
```

```
               MONTHLY BASIC NETWORK CONFIGURATION COSTS REPORT

                                                                PAGE   6

PROCESSOR          HARDWARE                            SOFTWARE
 NUMBER     RENTAL AND MAINTENANCE              RENTAL AND MAINTENANCE
                 COST($/MO.)                           COST($/MO.)

    1           $   14209.00                          $   580.00
    2           $1940710.00                           $ 1200.00
    3           $   70251.00                          $   714.00
    4           $   12439.00                          $   580.00
                ------------                          ----------
           TOTAL$2037609.00                      TOTAL$ 3074.00

  LINK            LINK COST
 NUMBER            ($/MO.)

   1              $ 580.13
   2              $ 539.22
   3              $ 526.84
   4              $ 520.48
   5              $ 635.29
                ----------
            TOTAL$2801.96

TOTAL MONTHLY COST OF BASIC NETWORK CONFIGURATION: $2043484.96
```

```
                     PROCESSOR OUTPUT REPORT

                        PROCESSOR NUMBER  1            PAGE    27

REPORT TIMES:

START REPORT TIME 5 UNITS
END REPORT TIME - 21599 UNITS

PROCESSOR UTILIZATION STATISTICS:       THE NUMBER OF JOB SEGMENTS
                                        DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.1299          UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS     UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 0.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY        NUMBER OF          MEAN                    MEAN
 OF JOB       FILE REQUESTS     THROUGHPUT                  CPU
                PROCESSED          TIME                    TIME

    1               4             122.25                  64.00
    2              21            3144.10                  15.86
    3              46            2519.15                  14.28
    4              70            3177.77                  12.93
    5              11            6111.91                  23.00
    6              13            5596.15                  29.54

JOB TYPE STATISTICS:

JOB TYPE          MEAN                NUMBER
  ID        THROUGHPUT TIME             OF
                PER JOB                JOBS

    1           9205.86                  7
    2           9234.54                 13
    3          10157.40                 15
    4          12769.23                 13

NOTE: TIME IS IN UNITS.
```

```
                I/O CHANNEL UTILIZATION STATISTICS

                              REPORT

                    PROCESSOR NUMBER  1                PAGE  28

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                  PERCENT                  NUMBER OF
 NUMBER                    UTILIZATION             I/O REQUESTS

    1                         0.129                    1384
    2                         0.011                     130
    3                         0.000                       1
    4                         0.000                       0
    5                         0.000                       0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                         PROCESSOR OUTPUT REPORT

                            PROCESSOR NUMBER  2             PAGE   29

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21599 UNITS

PROCESSOR UTILIZATION STATISTICS:         THE NUMBER OF JOBS SEGMENTS
                                          DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.1529            UNSVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS       UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 1.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY         NUMBER OF             MEAN                      MEAN
 OF JOB        FILE REQUESTS        THROUGHPUT                    CPU
                 PROCESSED             TIME                      TIME

    1                 7              7993.86                    11.00
    2                10                27.40                     5.20
    3               131              5821.66                    13.82
    4                64              2747.02                    11.56
    5                17              4369.75                    25.82
    6                15              2465.87                    12.27

JOB TYPE STATISTICS:

JOB TYPE             MEAN             NUMBER
  ID           THROUGHPUT TIME          OF
                   PER JOB             JOBS

    1             11466.14              28
    2             10744.25              24
    3             10152.61              23
    4             10610.87              24

NOTE: TIME IS IN UNITS.
```

```
                    I/O CHANNEL UTILIZATION STATISTICS

                                 REPORT

                          PROCESSOR NUMBER  2                PAGE   30

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                   PERCENT                  NUMBER OF
  NUMBER                    UTILIZATION               I/O REQUESTS

    1                          0.244                      3577
    2                          0.040                       601
    3                          0.003                        36
    4                          0.000                         0
    5                          0.000                         0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:
UNAVAILABLE TUPLES - 0
```

```
                        PROCESSOR OUTPUT REPORT

                          PROCESSOR NUMBER  3              PAGE   31

REPORT TIMES:

START REPORT TIME - 5 UNITS
END REPORT TIME - 21599 UNITS

PROCESSOR UTILIZATION STATISTICS:      THE NUMBER OF JOB SEGMENTS
                                       DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.1006         UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS    UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 2 JOBS
MEAN INPUT QUEUE LENGTH - 1.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY         NUMBER OF           MEAN                         MEAN
 OF JOB        FILE REQUESTS       THROUGHPUT                     CPU
                 PROCESSED           TIME                         TIME

    1               10             6249.10                        6.00
    2               35             4421.09                       16.23
    3               83             2159.08                       10.20
    4               35              748.80                       11.69
    5               10             1667.70                       14.10

JOB TYPE STATISTICS:

JOB TYPE              MEAN             NUMBER
  ID            THROUGHPUT TIME          OF
                   PER JOB              JOBS

    1              8371.00               9
    2              9488.27              11
    3             11843.14               7
    4             11749.46              13

NOTE: TIME IS IN UNITS.
```

```
                    I/O CHANNEL UTILIZATION STATISTICS

                                  REPORT

                         PROCESSOR NUMBER  3              PAGE    32

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                      PERCENT                    NUMBER OF
  NUMBER                       UTILIZATION                 I/O REQUESTS

     1                            0.118                      1429
     2                            0.012                       152
     3                            0.002                        27
     4                            0.000                         0

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                         PROCESSOR OUTPUT REPORT

                            PROCESSOR NUMBER  4            PAGE   33

REPORT TIMES:

START REPORT TIME 5 UNITS
END REPORT TIME - 21599 UNITS

PROCESSOR UTILIZATION STATISTICS:       THE NUMBER OF JOB SEGMENTS
                                        DELAYED DUE TO:

PROCESSOR UTILIZATION - 0.0578          UNAVAILABLE C/PS - 0
CURRENT INPUT QUEUE LENGTH - 0 JOBS     UNAVAILABLE MEMORY - 0
MAXIMUM INPUT QUEUE LENGTH - 1 JOBS
MEAN INPUT QUEUE LENGTH - 1.00 JOBS

JOB PRIORITY STATISTICS:

PRIORITY         NUMBER OF            MEAN                         MEAN
 OF JOB        FILE REQUESTS       THROUGHPUT                      CPU
                 PROCESSED            TIME                         TIME

    1                3               64.33                         1.33
    2               18               34.50                         1.78
    3               47               50.36                         6.57
    4               82             4720.83                         8.77
    5                6             3543.33                         4.50
    6                7             2352.43                        20.29
    7                3             8054.67                         5.33

JOB TYPE STATISTICS:

JOB TYPE           MEAN               NUMBER
   ID        THROUGHPUT TIME            OF
                 PER JOB               JOBS

    1           11827.64                11
    2           11150.64                14
    3            9744.56                 9
    4           10864.57                 7

NOTE: TIME IS IN UNITS.
```

```
                   I/O CHANNEL UTILIZATION STATISTICS

                                 REPORT

                        PROCESSOR NUMBER  4                  PAGE   34

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

I/O CHANNEL UTILIZATION STATISTICS:

I/O CHANNEL                        PERCENT                    NUMBER OF
 NUMBER                          UTILIZATION                 I/O REQUESTS

    1                               0.069                        608
    2                               0.004                         43
    3                               0.000                          4

THE NUMBER OF I/O REQUESTS DELAYED DUE TO:

UNAVAILABLE I/O CHANNELS - 0

THE NUMBER OF TRANSMISSION REQUESTS DELAYED DUE TO:

UNAVAILABLE TUPLES - 0
```

```
                    TRANSMISSION LINK STATISTICS REPORT

                                                            PAGE   35

REPORT TIMES:

START OF REPORT TIME - 5 UNITS
END OF REPORT TIME - 21599 UNITS

TUPLE REQUEST STATISTICS
FROM PROCESSOR N TO PROCESSOR M:

     N/M     1    2    3    4    5    6    7    8    9    10

      1      0  134    0  111    0    0    0    0    0     0
      2      0    0  160   87    0    0    0    0    0     0
      3      0    0    0  131    0    0    0    0    0     0
      4      0    0    0    0    0    0    0    0    0     0
      5      0    0    0    0    0    0    0    0    0     0

TUPLE UTILIZATION STATISTICS
FROM PROCESSOR N TO PROCESSOR M:

     N/M     1    2    3    4    5    6    7    8    9    10

      1        0.14 0.00 0.15 0.00 0.00 0.00 0.00 0.00 0.00
      2             0.21 0.13 0.00 0.00 0.00 0.00 0.00 0.00
      3                  0.23 0.00 0.00 0.00 0.00 0.00 0.00
      4                       0.00 0.00 0.00 0.00 0.00 0.00
      5                            0.00 0.00 0.00 0.00 0.00
      6                                 0.00 0.00 0.00 0.00
      7                                      0.00 0.00 0.00
      8                                           0.00 0.00
      9                                                0.00

TRANSMISSION PROCESSING STATISTICS:

PROCESSOR  TRANSMISSION     MAXIMUM       MEAN          MEAN
 NUMBER      REQUESTS    OUTPUT QUEUE  OUTPUT QUEUE  TRANSMISSION
            ORIGINATED      LENGTH        LENGTH      DELAY TIME

      1        113            3            1.03          1.58
      2        145            2            1.02          1.72
      3        142            2            1.01          2.46
      4        125            2            1.02          0.69

NOTE: TIME IS IN UNITS.
```

```
+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
=THE COMPUTER NETWORK SIMULATION WAS TERMINATED+
+          SUCCESSFULLY AT TIME  6.0000.       =
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

NETWORK RELIABILITY/AVAILABILITY REPORT

SIMULATION EXPERIMENT V

```
                NETWORK RELIABILITY/AVAILABILITY REPORT

                          NETWORK AS A WHOLE               PAGE   13

PESSIMISTIC NETWORK RELIABILITY IS   .99133E+00

MEAN NETWORK RELIABILITY IS   .99612E+00

PESSIMISTIC NETWORK AVAILABILITY IS  .81339E+00

MEAN NETWORK AVAILABILITY IS  .84327E+00
```

VITA

Hoyt M. Warren, Jr. was born on 13 February 1949 in Dothan, Alabama. He graduated from Auburn High School in Auburn, Alabama in 1967 and attended Auburn University, where he received, in 1971, a Bachelor of Science in Mathematics and a commission in the United States Air Force. He attended technical training at Keesler AFB, Mississippi and graduated in September of 1972. The next three years were spent as Chief of the Engineering and Installation Branch Air Force Cryptologic Depot. He entered the AFIT residence school in September 1975 and received the degree of Master of Science in Computer Science in March 1977.

Permanent Address: 884 Terrace Acres
Auburn, Alabama 36830

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER GCS/EE/77-3	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A GENERAL COMPUTER NETWORK SIMULATION MODEL		5. TYPE OF REPORT & PERIOD COVERED AFIT Thesis
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Hoyt M. Warren, Jr., Capt, USAF		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology (AFIT-EN) WPAFB OH45433		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Data System Design Center (LGLA/AFDSDC) Gunter AFB, AL 36117		12. REPORT DATE March 1977
		13. NUMBER OF PAGES 312
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Approved for public release;
IAW AFR 190-17
Jerral F. Guess, Captain, USAF
Director of Information

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

computer network
simulation
modeling
performance evaluation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The emphasis of this investigation is on the development of a general computer network simulation model for the evaluation of alternative computer network configurations. The simulation model allows a computer network and its workload to be characterized and then allows selected performance measures to be made.

The various phases of the development process are discussed. In the initial phase, symbolic system modeling and simulation techniques are

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

surveyed. This discussion dwells heavily upon discrete-event modeling and simulation techniques. Next, the component architectures and essential performance measures of computer networks are investigated. Then, a computer network model is presented. The features of the network model are discussed and a discription of job flow is given. This discussion describes the characterization and interrelationships of the various model variables and the algorithms governing their behavior. The implementation of the computer network model is then described and through a series of simulation experiments the behavior of the computer network simulation model(implementation) is compared with the behavior predicted by the network model.

The simulation experiments indicated that the simulation model was an accurate reflection of the computer network model. Several areas of potential improvement to the model and simulation are indicated and discussed.